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

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Hierarchical Scaled dot-product attention for Insect Pest Recognition

Vu Thinh Doan1, Hoang Thanh Le2, and Son Lam Phung3

1Faculty of Information Technology, Nha Trang University, Khanh Hoa, Vietnam

2School of Computing and Information Technology, University of Wollongong

3Electrical Engineering Department, University of Colorado, Boulder, CO 80309 USA

Corresponding author: First A. Author (e-mail: thinhdv@ntu.edu.vn).

ABSTRACT Insect pests are the primary factor effecting agriculture crops. Therefore, accurately identifying various types of crop insects is an important task for biosecurity and agriculture. To address this, we propose Hierarchical Scaled Dot-Product Attention (HSDPA), a method leverages scaled dot-product attention to focus on important features of pest images at different levels. We also introduce Spatial Attention (SPA), a method that extracts features along both horizontal and vertical dimensions of the input feature map, combined with multiple Max Pooling modules with different kernel sizes to extract feature information across various scales. Comparative experiments prove the effectiveness of our approach. Our proposal achieves 68.6%mAP0.5 on the IP102 dataset. Our code is available at: https://github.com/thinhdoanvu/HSPDA

INDEX TERMS Hierarchical Scaled Dot-Product Attention, Pest detection, Spatial Attention, IP102

1. INTRODUCTION

Pests are a major threat to agricultural production, causing significant economic and environmental losses [1]. Detecting pests is essential, as it helps farmers quickly and accurately identify harmful species, allowing them to take timely measures to protect crops. Most current methods rely on manual inspection [2], pheromone traps [3], and plant and soil analysis [4]. These methods are straightforward, they are often slow and labor-intensive.

With advances in deep learning, various studies have focused on developing automated systems for detecting agricultural pests [5]. Deep learning-based methods can automatically extract key features from images, enabling real-time pest detection [6]. Deep learning-based pest detection methods can be categorized into one-stage and two-stage approaches. Two-stage methods, like R-CNN, Faster R-CNN, and Mask R-CNN work in two steps. First, they identify areas in an image that might have objects, and then they classify those areas. This method is usually very accurate but can be slow and requires a lot of computational time. In contrast, one-stage methods, such as YOLO (You Only Look Once) and SSD (Single Shot MultiBox Detector), perform detection and classification in one step. This makes them faster and more efficient for real-time use. However, they may not be as accurate as two-stage methods.

The deep learning-based object recognition methods mentioned above have achieved impressive results. However, they have limited applications in pest monitoring due to the unique characteristics of pest detection, which differ from conventional object detection and classification tasks [7].

In real-world field images, pests are usually small objects surrounded by complex environments. This can easily confuse recognition systems, as they may misinterpret the background while identifying key points [8]. Additionally, different pest types often exhibit similar physical features, and even the same pest species can appear in various forms, such as larvae, eggs, pupae, and adults [9].

To address these gaps, we propose a novel one-stage method, Hierarchical Scaled dot-product attention. This approach utilizes Transformer architecture to effectively detect pests from start to finish in a single process. To enhance target detection accuracy, we introduce Hierarchical Scaled dot-product attention (HSDPA) and Spatial Attention (SPA) modules into the network. The HSDPA module enhances feature extraction and encoding. SPA extract feature information at different scales. In summary, our main contributions are as follows:

* We introduce Hierarchical Scaled dot-product attention (HSPDA) and Spatial Attention (SPA) modules based on self-attention to enhance network information extraction and multiscale feature extraction.
* Our approach shows a notable improvement in detection accuracy compared to the current leading methods in the field.

The remaining paper is organized into the following sections: the existing works are discussed in Section II, and the proposed work in Section III. The results and conclusion are demonstrated in Sections IV and V, respectively.

1. RELATED WORK

This section provides an overview of relevant research. Subsection II-A examines object detection techniques based on deep learning. Subsection II-B focuses on deep learning approaches applied to pest detection..

1. OBJECT DETECTION METHODS

Traditional object detection methods usually consist of three stages. (1) Candidate Region Generation: Initially, candidate regions are identified in an image, often using a sliding window technique to locate potential objects; (2) Feature Extraction: Features from these candidate regions are extracted using methods like SIFT [10] or HOG [11]. The effectiveness of feature extraction significantly influences the performance of the next stage; (3) Classification: A trained classifier, such as Support Vector Machine (SVM) or AdaBoost [12], is then applied to classify the detected objects. While the three-stage method is effective in handling variations in pedestrian poses and achieving higher detection accuracy in complex scenes, it has notable limitations. This approach suffers from high computational demands, resulting in slower matching speeds, which negatively affects its real-time performance [13].

Two-stage object detectors operate in two steps: they first generate region proposals, or potential bounding boxes, where objects may be located. Then, a deep neural network extracts features from these regions, and a classification network assigns object classes and refines bounding boxes based on these features. This approach combines localization and classification, offering precise object detection [13]. R-CNN, developed in 2014 by Girshick et al. [14], uses selective search, a greedy approach to generate around 2,000 region proposals. Each proposal is passed through a Convolutional Neural Network (CNN), such as AlexNet or VGG16, to extract features, which are then classified by an SVM. A bounding box further refines localization for more accurate detection. R-CNN faces significant limitations due to its slow training process. Each component of the model, including the CNN, SVM classifier, and bounding box, must be trained independently. This makes the model less efficient and not ideal for applications require real-time processing. Fast R-CNN is an algorithm developed in 2015 by Girshick [15], enhanced the original R-CNN framework. It integrates the region proposal generation step by utilizing a Region Proposal Network (RPN) that generates proposals more efficiently. The RPN shares convolutional features with the detection network, significantly reducing computation time. Instead of processing each region proposal separately, Fast R-CNN passes the entire image through the CNN once to extract a shared feature map, from which features for each proposal are then extracted. Faster R-CNN, developed by Ren et al. in 2015 [16], is an enhanced version of the original R-CNN and Fast R-CNN models. This algorithm enhances the two methods previously by integrating region proposal generation into the network architecture. Faster R-CNN replaces the traditional selective search algorithm with a convolutional network that generates region proposals directly within the model.

One-stage object detectors use a single neural network to scan the entire image and directly predict object locations. By dividing the image into a grid, each cell handles detection for objects centered within it. The network assigns anchor boxes of different sizes, scoring them for object likelihood and refining bounding box predictions. Non-maximum suppression is then applied to retain high-scoring detections, eliminating the need for a separate region proposal step [17]. RetinaNet is a one-stage object detection model introduced by Facebook AI Research in 2017, designed to improve detection accuracy, especially for small and challenging objects. It uses a specialized "focal loss" function to address the challenge of class imbalance between background and foreground objects. Focal loss reduces the impact of easy, well-classified examples (often background) and focuses on harder, less-detected objects, allowing it to achieve high accuracy comparable to two-stage detectors while maintaining faster inference speeds. RetinaNet’s architecture includes a Feature Pyramid Network (FPN) for extracting features at multiple scales, along with detection heads that predict class probabilities and bounding boxes [18].

YOLO (You Only Look Once) is an innovative object detection model that analyzes images with just one pass through the neural network. The original YOLO model, introduced in 2015 by Joesph Redmon et al. [19], employs an "s x s" grid overlay on the image. Each grid cell detects an object if the object’s center lies within it, enabling other cells to disregard the object in cases of overlap. For each cell, YOLO predicts "B" bounding boxes, providing dimensions and confidence scores [20]. Subsequently, YOLOv2 [21] uses the Darknet-19 framework as its backbone, which consists of 19 convolutional layers and 5 max-pooling layers. Additionally, YOLOv2 introduced anchor boxes and removed fully-connected layers to improve efficiency. Joseph Redmon et al. introduced YOLOv3 [22] in 2018, utilizing an extended backbone with 53 convolutional layers and incorporating residual connections to enhance performance. The notable improvement is the modified Spatial Pyramid Pooling (SPP) block within the backbone, which allows YOLO to achieve a broader receptive field and improve feature extraction capabilities. Alexey Bochkovskiy and his team introduced YOLOv4 [23] in April 2020. The integration of Cross Stage Partial (CSP Darknet53), an improved SPP structure [24], Path Aggregation Network (PANet) architecture [25], Cross-Iteration Batch Normalization (CBN) integration [26], and segment-anything model (SAM) inclusion [27] led to a more efficient and robust object detection model. Glenn Jocher introduced YOLOv5 in 2020, developed by Ultralytics, which takes a different approach by using PyTorch instead of Darknet for its framework. YOLOv5 incorporates a Cross Stage Partial (CSP) Net, based on the ResNet architecture, utilizing partial connections between stages to improve network efficiency. Additionally, the CSPNet is enhanced with several SPP blocks, enabling feature extraction at multiple scales. The Meituan Vision AI Department introduced YOLOv6 in September 2022. It features the new CSPDarknet backbone, which improves efficiency and speed compared to YOLOv4 and YOLOv5. A key enhancement is the integration of a feature pyramid network (FPN), which enhances detection accuracy by expanding the range of feature scales. Wang etal. [28] introduced YOLOv7 in 2022, using the Efficient Layer Aggregation (ELAN) network is utilized to enhance training while preserving the integrity of the original gradient path. Additionally, auxiliary heads are incorporated into the shallow layers of the network to boost performance. In January 2023, Ultralytics launched YOLOv8, marking a significant development in the field of computer vision. YOLOv8 features advanced backbone and neck architectures for improved feature extraction and detection performance. Released in February 2024, YOLOv9 [29] introduces two major innovations: (1) the Programmable Gradient Information (PGI) framework helps retain important gradients and features across layers, minimizing issues like vanishing gradients or gradient saturation; (2) the Generalized Efficient Layer Aggregation Network (GELAN) improves information flow across network stages, enhancing feature fusion and detection performance, particularly for small objects. It also supports scalable architectures, making YOLOv9 adaptable to various hardware and deployment setups. YOLOv10 was released in May 2024. Its architecture builds on previous YOLO models while introducing several key innovations. Notably, YOLOv10 uses an enhanced CSPNet backbone, improving gradient flow and reducing computational redundancy. The neck utilizes PAN (Path Aggregation Network) layers to effectively fuse features at multiple scales. For training, YOLOv10 employs a One-to-Many Head, which generates multiple predictions per object to improve learning accuracy. During inference, the model uses a One-to-One Head, which produces the best prediction for each object, eliminating the need for Non-Maximum Suppression (NMS), thereby reducing latency and improving efficiency. YOLOv11, released in October 2024, is an advancement over YOLOv8. A key update in YOLOv11 is the replacement of the C2f block in the neck with a C3k2 block, which improves processing speed by using two smaller convolution operations instead of one large kernel. Additionally, YOLOv11 introduces the C2PSA (Cross Stage Partial with Spatial Attention) block, which enhances the model’s ability to focus on important areas in feature maps, boosting detection accuracy by improving the spatial pooling of features.

The DEtection TRansformer (DETR), introduced by Carion et al. [30], was the first end-to-end object detection framework to implement a transformer encoder-decoder architecture. The model directly predicts objects after passing the input through an encoder-decoder backbone (such as ResNet) for feature extraction. This novel approach removes the need for traditional components like region proposals and non-maximum suppression, which are commonly used in other detection frameworks like Faster R-CNN. However, DETR's high computational costs have limited its real-time applicability, prompting further research into optimizing transformer-based models for practical use.

1. PEST DETECTION METHODS

Use one space after periods and colons. Hyphenate complex modifiers: “zero-field-cooled magnetization.” Avoid dangling participles, such as, “Using (1), the potential was calculated.” [It is not clear who or what used (1).] Write instead, “The potential was calculated by using (1),” or “Using (1), we calculated the potential.”

Use a zero before decimal points: “0.25,” not “.25.” Use “cm3,” not “cc.” Indicate sample dimensions as “0.1 cm × 0.2 cm,” not “0.1 × 0.2 cm2.” The abbreviation for “seconds” is “s,” not “sec.” Use “Wb/m2” or “webers per square meter,” not “webers/m2.” When expressing a range of values, write “7 to 9” or “7-9,” not “7~9.”

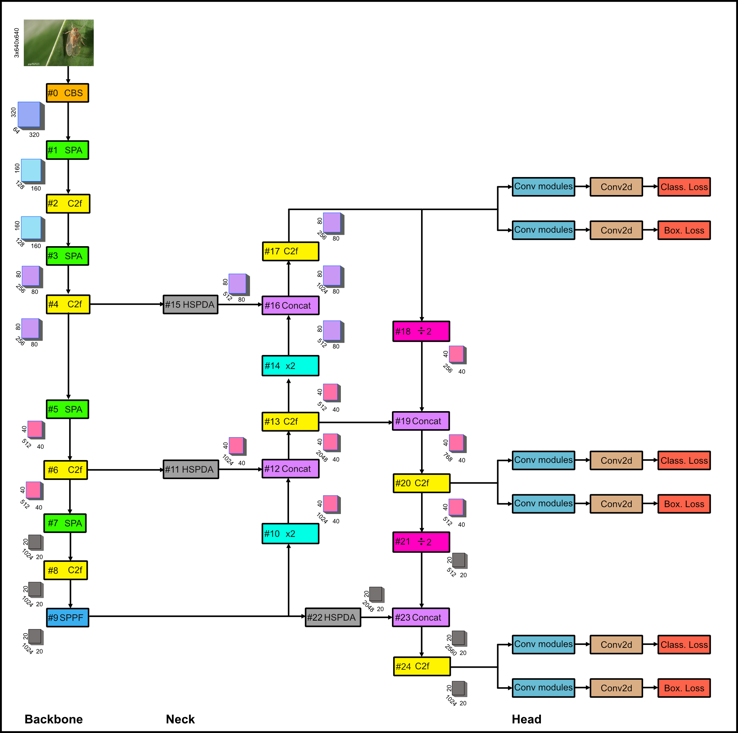
A parenthetical statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.) In American English, periods and commas are within quotation marks, like “this period.” Other punctuation is “outside”! Avoid contractions; for example, write “do not” instead of “don’t.” The serial comma is preferred: “A, B, and C” instead of “A, B and C.”

If you wish, you may write in the first person singular or plural and use the active voice (“I observed that ...” or “We observed that ...” instead of “It was observed that ...”). Remember to check spelling. If your native language is not English, please get a native English-speaking colleague to carefully proofread your paper.

METHOD

In this section, we introduce the proposed main method for pest detection. Section III-A provides an overview of our architecture. Section III-B discusses modules designed to enhance feature encoding, including CBS, C2f, SPPF, and SPA. Section III-C presents the Hierarchical Scaled Dot-Product Attention (HSDPA) module, which improves feature fusion. Section III-D describes the head block.

1. OVERVIEW OF ARCHITECTURE

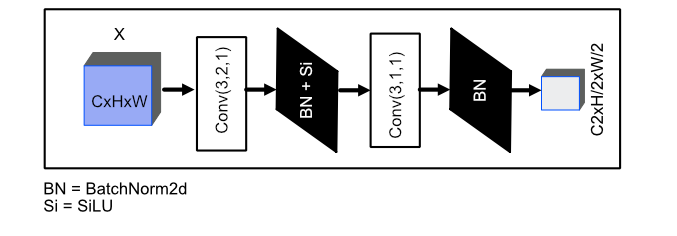


1. Improved YOLOv8 network structure.

As shown in Figure 1, the overall architecture based on YOLOv8 consists of three main elements: the backbone, neck, and detection layers. Each of these components will be introduced in the subsequent sections. YOLOv8 is a version within the YOLO series of object detection algorithms, created by Ultralytics. This version is the latest widely available implementation, making it reliable for real-world applications. YOLOv8 achieves a balanced trade-off between speed and efficiency for various application tasks compared to more advanced versions like YOLOv9, YOLOv10, and YOLOv11 [31]. The model uses an optimized backbone network to reduce computational requirements and improve target detection at various scales through the use of Feature Pyramid Network (FPN) and Spatial Pyramid Pooling Fast (SPPF) modules. However, YOLOv8 still has some limitations in certain application scenarios. These include challenges such as small objects in complex scenes [32], poor image quality [33], an imbalance in pest categories [34], high similarity in the appearance features of different pest species [6]. To address the problems mentioned above, we enhanced the YOLOv8 by incorporating the Spatial Attention (SPA) (Fig. 5) and the Hierarchical Scaled dot-product attention (HSDPA) (Fig. 7) modules.

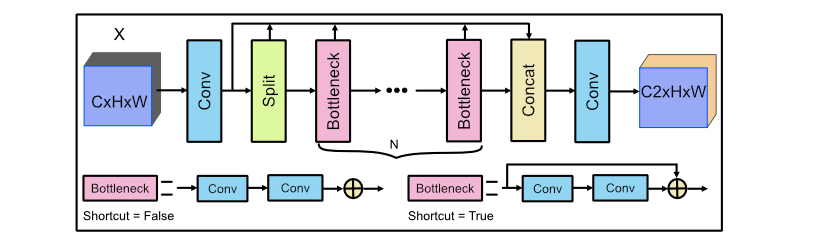
1. BACKBONE

For the backbone, we use the CBS module (Fig. 2) to extract features, followed by down-sampling to reduce the spatial dimensions.



1. CBS module with Convolutional layers, Batch Normalization, and SiLU activation for reducing spatial dimensions while extracting essential features.

The CBS module is designed to enhance feature extraction and stability in neural networks through a structured sequence of layers. It starts with a convolutional layer using a 3x3 kernel, stride of 2, and padding of 1, which reduces spatial dimensions while extracting essential features from the input. This is followed by BatchNorm2d layers to normalize the output, helps the network to learn better and differentiate between essential features in the image while avoiding less relevant activations. Next, the Sigmoid Linear Unit (SiLU) activation function introduces non-linearity, enabling the model to capture more complex patterns within the data. A second convolutional layer, also with a 3x3 kernel but with a stride of 1 and padding of 1, further refines and extracts detailed features while preserving the spatial dimensions.



1. The structure of the C2f block allows for adjustments to the accumulation count 𝑁 of Bottleneck blocks as well as the shortcut connections.

The Cross Stage Partial Bottleneck with 2 convolutions (C2f) block (Fig. 3) starts with a convolutional layer, followed by splitting the resulting feature map into two paths. One path is processed through the Bottleneck block, while the other is directly sent to the Concat block. The number of Bottleneck blocks in the C2f block is controlled by the depth\_multiple parameter of the model (Table. I). The bottleneck block is composed of a series of convolutional layers structured similarly to a ResNet block. After processing, the feature maps from the Bottleneck blocks are merged with the split feature map and passed through a final convolutional layer. This design incorporates cross-stage connections and feature reuse strategies, enabling enhanced gradient flow and improved network efficiency. Additionally, the Bottleneck structure leverages residual connections to preserve original performance while reducing number of parameters. In the YOLOv8 variant, three key parameters influence its configuration: depth\_multiple (d), width\_multiple (w), and max\_channels (mc). The depth\_multiple controls the number of Bottleneck blocks within the C2F block, while the width\_multiple and max\_channels determine the number of output channels. The calculation of output channel values (Eq. 1) bases on the parameters width\_multiple and max\_channels [35], as follows:

(1)

For instance, in the backbone's initial CBS block (Fig. 1), the output channel is 64. With the width\_multiple (w) set to 1 and the max\_channels (mc) set to 1024, the computation proceeds as follows:

TABLE I

YOLOv8 Variants

|  |  |  |  |
| --- | --- | --- | --- |
| types | depth\_multiple | width\_multiple | max\_channels |
| n | 0.33 | 0.25 | 1024 |
| *s* | 0.33 | 0.5 | 1024 |
| *m* | 0.67 | 0.75 | 768 |
| *l* | 1.0 | 1.0 | 512 |
| *x* | 1.0 | 1.25 | 512 |

Accuracy, speed, and model size are influenced by the model types, depth\_multiple, width\_multiple, and max\_channels parameters.

depth-multiple (d) controls the number of Bottleneck blocks within the C2f block.

width\_multiple (w) adjusts the number of channels in the convolutional layers.

max\_channels (mc) defines the maximum allowable number of channels in the network.

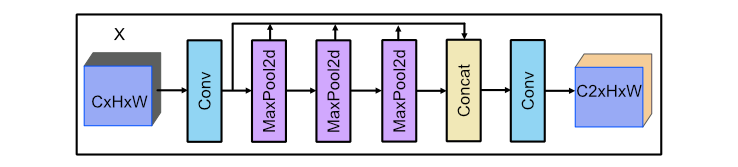
n represents the smallest size, offering the fastest inference but with the lowest accuracy.

s provides a good balance between speed and accuracy, making it suitable for general use.

m offers higher accuracy than smaller models, while still maintaining moderate inference speed.

l delivers the highest accuracy, but with slower inference.

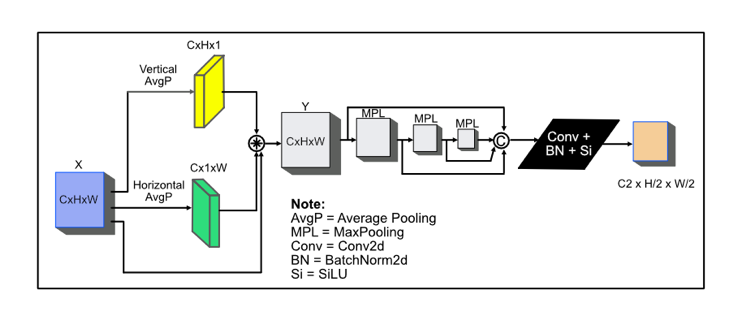
x excels in accuracy, making it ideal for resource-intensive applications.



1. SPPF structure diagram.

Next is the SPPF block, an important part of the YOLOv8 architecture shown in Fig. 4. SPPF stands for Spatial Pyramid Pooling Fast, first introduced in YOLOv5. SPPF module is an optimized Spatial Pyramid Pooling (SPP) version, which was originally implemented in YOLOv3, with the same mathematical functionality but fewer floating-point operations (FLOPs). The SPPF block begins with convolutional layers, followed by three MaxPool2d layers. Each resulting feature map from the max pooling layers is concatenated at the end and fed into a final convolutional layer. SPPF use only a single fixed-size kernel for pooling, reducing the number of computations required.

Inspired by Coordinate Attention (CA) [35] and SPP Block [22], the Spatial Attention (SPA) module was developed (Fig. 5). Our module integrates spatial attention mechanisms and multi-scale pooling to enhance feature extraction. It starts by performing global average pooling separately along the horizontal and vertical dimensions. This process refines the feature representation by highlighting the most important spatial relationships in each direction. The output of the *c-th* channel at height ℎ and width 𝑤 is described in (Eq. 2, 3). Following this, three max-pooling layers with different kernel sizes are employed to capture information at different scales. Each max-pooling operation reduces the spatial dimensions with a stride, while padding ensures that the outputs are aligned for concatenation. Feature maps from the three pooling layers are then concatenated along the channel dimension to form a single multi-scale representation. A convolutional layer is subsequently applied to reduce the number of channels to C2, followed by batch normalization to stabilize the output and improve convergence. Finally, a SiLU activation function introduces non-linearity, enabling the network to model complex relationships in the data.

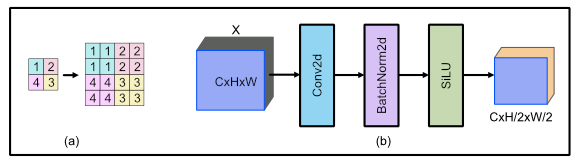


1. Spatial Attention (SPA) block illustrated.

(2)

(3)

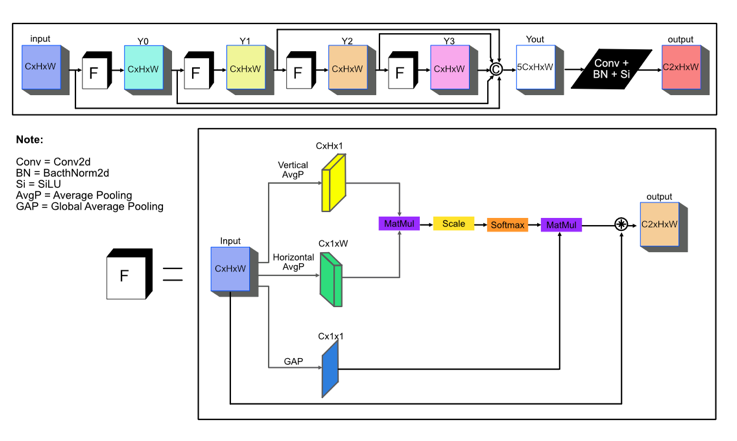
1. NECK



1. Upsampling by nearest neighbor (a) and Downsampling (b).

After exiting the SPPF block, the output is passed to the blocks in the neck section of the model. The first component is an up-sampling layer (Fig. 6-a), which simply increases the feature map by a factor of two without making any changes to the output channel.

The HSDPA module is a neural network layer that applies the Scale Dot Product (SDP) attention mechanism multiple times on the input tensor to capture different feature representations. Initially, the input tensor is passed through the SDP module 4 times (F blocks), generating four different outputs (Y0~3). These outputs and input are then concatenated along the channel dimension to form a tensor with 5 times the original number of channels. This concatenated tensor is processed by a 1x1 convolutional layer, which reduces the channel dimension to a specified value (C2). After the convolution, batch normalization is applied to normalize the activations across the batch, improving training stability. Finally, a SiLU activation function is applied to introduce non-linearity into the model, allowing it to learn more complex relationships in the data. This module is designed to enhance feature representations through attention mechanisms and can be get more useful information from different parts of the input.



1. Hierarchical Scaled dot-product attention (HSDPA) module structure diagram.

The Scaled Dot-Product Attention mechanism, as utilized in Transformer architectures [36], computes attention by leveraging the relationships between different spatial and channel dimensions of the input feature map. Given an input tensor 𝑥∈𝑅𝐵×𝐶×𝐻×𝑊, where 𝐵 denotes the batch size, 𝐶 represents the number of channels, and 𝐻 and 𝑊 are the height and width of the feature map, the mechanism computes attention separately across the height (), width (), and channel () dimensions (Eq. 4-6).

(4)

(5)

(6)

After obtaining the attention maps for height, width, and channel dimensions, the attention scores are computed via matrix multiplication between the height attention query tensor 𝑄 and the width attention key tensor 𝐾. The attention score matrix 𝑆 is given by:

(7)

Subsequently, the scores are scaled by the square root of the depth to prevent large gradient values, resulting in the scaled scores :

(8)

The scaled attention scores are passed through a SoftMax function to obtain the attention weights :

(9)

These attention weights are used to weight the value tensor , obtained from the channel attention mechanism.

The final output is computed by performing matrix multiplication between the attention weights and the value tensor :

(10)

The output tensor is reshaped to match the original spatial dimensions, and then element-wise multiplied with the input tensor to refine the output:

(11)

1. HEAD

For the backbone, YOLOv8 adopts an anchor-free approach, predicting the center of an object directly rather than calculating the offset relative to a predefined anchor box. Anchor-free detection minimizes the number of box predictions, accelerating complex post-processing steps that filter candidate detections after inference. The detection block processes feature maps from the neck to produce two outputs: (1) Predicts the object class probabilities for each grid cell; (2) Predicts the coordinates and dimensions of bounding boxes for each grid cell. The class loss is determined using binary cross-entropy, which evaluates the confidence scores associated with each predicted bounding box. Meanwhile, the box loss is calculated as the mean squared error (MSE) between the predicted and ground truth bounding box parameters, considering object spatial locations, shapes, and varying aspect ratios. The overall detection loss is typically a weighted sum of the class and box losses (Eq. 12)

(12)

Where:

: Weight for class loss

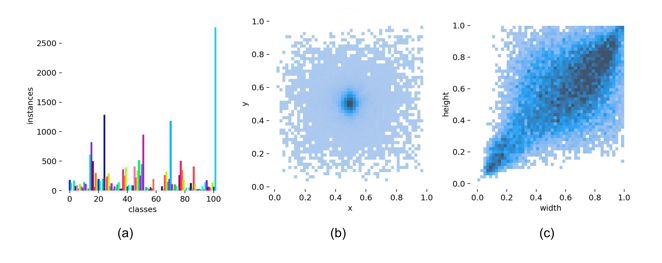
: Weight for box loss

EXPERIMENTS AND RESULTS

In this section, we outline the experimental methodology and examine the results. Subsection IV-A provides an overview of the proposed IP102 dataset. Subsection IV-B explains the experimental setup in detail. Subsection IV-C outlines the metrics used for evaluating the methods. Subsection IV-D focuses on the ablation studies conducted during development. Lastly, Subsection IV-E presents a comparative analysis of the quantitative and qualitative outcomes of various approaches.

1. IP102 DATASET

The IP102 dataset comprises 75,222 RGB images spanning 102 pest categories, with 18,983 images specifically allocated for detection tasks. The dataset is further divided into a training set containing 15,178 images and a testing set with 3,798 images, both annotated with bounding boxes. There are 8 super classes. Rice, Corn, Wheat, Beet, Alfalfa belong to Field Crop (FC) and Vitis, Citrus, Mango belong to Economic Crop (EC). These images, largely close-ups of individual pests with complex backgrounds, varying lighting conditions, and a wide range of pest species. This dataset presents various factors that impact and challenge the effectiveness of classification and image recognition models. Firstly, this dataset demonstrates a notable imbalance, with certain classes being heavily overrepresented while others are underrepresented (Fig. 8-a). Secondly, the heatmap in Fig. 8-b reveals a concentration of object centers near the image center, indicating a potential bias in object placement that could affect model performance. Thirdly, Fig. 8-c illustrates the distribution of bounding box dimensions, where the normalized width and height suggest that the majority of objects are small, posing additional challenges for detection. Lastly, pests experience various growth stages during their lifecycle, with each stage displaying distinct external characteristics. These differences lead to significant intra-class variations, making it difficult to maintain consistent detection performance. (Fig. 9).



1. Training set statistics. (a) histogram of class distribution, (b) heatmap of labels position, and (c) heatmap of labels dimension.



1. Different forms of rice leaf caterpillar images.
2. IMPLEMENTATION DETAILS

We designed and conducted experiments on our method using the Ultralytics framework. For optimization during training, we employed the SDG optimizer with a learning rate of 0.01 and momentum of 0.9. The model input size was set to 640×640 pixels, with a batch size of 32. We trained the model for 200 epochs, selecting the best-performing model based on the validation set as the final version. All training and testing were performed on an NVIDIA GeForce RTX 4090 GPU.

1. EVALUATION INDICATORS

The evaluation metrics for target detection employed in this study consist of Model Parameters (Params), Giga Floating Point Operations (GFLOPs), average accuracy (mAP50 and mAP50-95) as defined in equations 13 - 16.

* *P* evaluates the model's accuracy by calculating the proportion of true pests among all samples the model identifies as pests. The formula is as follows:

(13)

Where:

TP represents instances where the model accurately identifies a positive sample as positive.

FP represents instances where the model mistakenly identifies a negative sample as positive.

* *R* indicates the proportion of images that actually contain pests and are correctly identified by the model. The formula is as follows:

(14)

Where:

FN refers to instances where the model mistakenly classifies a sample that should be positive (i.e., contains pests) as negative (i.e., not containing pests).

* Mean Average Precision (mAP) is a performance metric commonly used to assess object detection models. mAP score is determined by averaging the Average Precision (AP) across all classes and/or various Intersection over Union (IoU) thresholds, depending on the specific detection task. The formula for mAP is:

(15)

Where:

N is the total number of classes

APk is the Average Precision for class k

mAP0.5: This is the average precision calculated at an IoU threshold of 0.5.

mAP0.5:0.95: This is the average precision calculated across multiple IoU thresholds ranging from 0.5 to 0.95, with values computed at intervals of 0.05.

1. ABLATION STUDY

We performed an ablation study using the IP102 dataset to assess how key components affect our model's performance. As shown in Table II, we compared the performance of different YOLOv8 model sizes. YOLOv8 offers five versions: Nano (N), Small (S), Medium (M), Large (L), and Extra Large (X). The results demonstrate that prediction accuracy improves as the model size increases.

TABLE II

Experimental Results of YOLOv8 with Different Model Sizes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Models | mAP50 | mAP50:95 | #Params | GFLOPs |
| n | 65.8 (+0.0) | 41.7 (+0.0) | 3.157.200 | 8.9 |
| *s* | 68.3 (+2.5) | 43.1 (+1.4) | 11.166.560 | 28.8 |
| *m* | 67.9 (+2.1) | 42.9 (+1.2) | 25.902.640 | 79.3 |
| *l* | 68.4 (+2.6) | 44.3 (+2.6) | 43.691.520 | 165.8 |
| *x* | 69.5 (+3.3) | 44.4 (+2.7) | 68.229.648 | 258.5 |

Table II demonstrates that the YOLOv8-S model achieves a 2.5% increase in detection mAP50 accuracy. From the smallest model to the extra large model, there is a marginal improvement of 1.2% in detection mAP50. However, this improvement comes at the cost of significantly higher computational complexity and an increased number of parameters, which can greatly impact inference speed. As a result, YOLOv8-S and YOLOv8-X were chosen as the baseline models for this study, as they provide a balance between prediction accuracy and model efficiency. Specifically, the values for depth, width, and max\_channels were set to 0.33, 1.00, and 1024, respectively.

TABLE III

Number of levels (K) for Hierarchical Scale-dot Product Attention

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| k | mAP50 | mAP50:95 | #Params | GFLOPs |
| 2 | 68.5 | 43.3 | 74,155,723 | 172.5 |
| *3* | 69.0 | 44.0 | 76,908,235 | 186.3 |
| ***4*** | **69.4** | **44.1** | **79,600,609** | **199.9** |
| *5* | 69.3 | 44.1 | 82,413,259 | 214.1 |
| *6* | 68.7 | 44.2 | 85,165,771 | 227.9 |

Table III?

1. DETECTION PERFORMANCE

We performed an ablation study using the IP102 dataset to

TABLE IV

Comparative Experiments of Different Target Detection Models

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Methods | mAP50 | mAP50:95 | #Params | GFLOPs |
| **Two-stage Methods** | | | | |
| Faster R-CNN |  |  |  |  |
| Dynamic R-CNN |  |  |  |  |
| Cascade R-CNN |  |  |  |  |
| EfficientDet |  |  |  |  |
| RetinaNet |  |  |  |  |
| **Transformer-based Methods** | | | | |
| DETR |  |  |  |  |
| RT-DETR |  |  |  |  |
| **One-stage Methods** | | | | |
| SSD |  |  |  |  |
| YOLOv9-T[29] |  |  | 2,128,720 | 8.5 |
| YOLOv9-S[29] | 66.1 | 43.2 | 7,318,368 | 27.6 |
| YOLOv9-M[29] | 68.7 | 43.6 | 20,216,160 | 77.9 |
| YOLOv9-C[29] | 68.3 | 42.5 | 25,590,912 | 104.0 |
| YOLOv9-E[[1]](#footnote-1)[29] | x | x | 58,206,592 | 193.0 |
| YOLOv10-N[37] | 63.2 | 40.3 | 2,839,772 | 9.1 |
| YOLOv10-S[37] | 66.9 | 43.3 | 8,145,300 | 25.2 |
| YOLOv10-M[37] | 66.9 | 43.4 | 16,602,244 | 64.6 |
| YOLOv10-B[37] |  |  | 20,608,308 | 99.5 |
| YOLOv10-L[37] | 62.5 | 40.0 | 25,922,612 | 128.1 |
| YOLOv10-X[37] |  |  |  |  |
| YOLOv11-N[38] |  |  | 2,624,080 | 6.6 |
| YOLOv11-S[38] |  |  | 9,458,752 | 21.7 |
| YOLOv11-M[38] |  |  | 20,114,688 | 68.5 |
| YOLOv11-L[38] | 67.3 | 40.2 | 25,372,160 | 87.6 |
| YOLOv11-X[38] |  |  | 56,966,176 | 196.0 |
| **Ours** | **69.4** | **43.9** | **79,660,747** | **200.2** |

SOME COMMON MISTAKES

The word “data” is plural, not singular. The subscript for the permeability of vacuum µ0 is zero, not a lowercase letter “o.” The term for residual magnetization is “remanence”; the adjective is “remanent”; do not write “remnance” or “remnant.” Use the word “micrometer” instead of “micron.” A graph within a graph is an “inset,” not an “insert.” The word “alternatively” is preferred to the word “alternately” (unless you really mean something that alternates). Use the word “whereas” instead of “while” (unless you are referring to simultaneous events). Do not use the word “essentially” to mean “approximately” or “effectively.” Do not use the word “issue” as a euphemism for “problem.” When compositions are not specified, separate chemical symbols by en-dashes; for example, “NiMn” indicates the intermetallic compound Ni0.5Mn0.5 whereas “Ni–Mn” indicates an alloy of some composition NixMn1-x.

Be aware of the different meanings of the homophones “affect” (usually a verb) and “effect” (usually a noun), “complement” and “compliment,” “discreet” and “discrete,” “principal” (e.g., “principal investigator”) and “principle”

GUIDELINES FOR GRAPHICS PREPARATION   
AND SUBMISSION

1. TYPES OF GRAPHICS

The following list outlines the different types of graphics published in IEEE journals. They are categorized based on their construction, and use of color / shades of gray:

1. Color/Grayscale figures

Figures that are meant to appear in color, or shades of black/gray. Such figures may include photographs, illustrations, multicolor graphs, and flowcharts.

1. Line Art figures

Figures that are composed of only black lines and shapes. These figures should have no shades or half-tones of gray, only black and white.

1. Author photos

Head and shoulders shots of authors that appear at the end of our papers.

1. Tables

Data charts which are typically black and white, but sometimes include color.

TABLE I

Units for Magnetic Properties

|  |  |  |
| --- | --- | --- |
| Symbol | Quantity | Conversion from Gaussian and  CGS EMU to SI a |
| Φ | magnetic flux | 1 Mx → 10−8 Wb = 10−8 V·s |
| *B* | magnetic flux density,  magnetic induction | 1 G → 10−4 T = 10−4 Wb/m2 |
| *H* | magnetic field strength | 1 Oe → 103/(4π) A/m |
| *m* | magnetic moment | 1 erg/G = 1 emu  → 10−3 A·m2 = 10−3 J/T |
| *M* | magnetization | 1 erg/(G·cm3) = 1 emu/cm3  → 103 A/m |
| 4π*M* | magnetization | 1 G → 103/(4π) A/m |
| σ | specific magnetization | 1 erg/(G·g) = 1 emu/g → 1 A·m2/kg |
| *j* | magnetic dipole  moment | 1 erg/G = 1 emu  → 4π × 10−10 Wb·m |
| *J* | magnetic polarization | 1 erg/(G·cm3) = 1 emu/cm3  → 4π × 10−4 T |
| χ*,* κ | susceptibility | 1 → 4π |
| χρ | mass susceptibility | 1 cm3/g → 4π × 10−3 m3/kg |
| μ | permeability | 1 → 4π × 10−7 H/m  = 4π × 10−7 Wb/(A·m) |
| μr | relative permeability | μ → μr |
| *w, W* | energy density | 1 erg/cm3 → 10−1 J/m3 |
| *N, D* | demagnetizing factor | 1 → 1/(4π) |

Vertical lines are optional in tables. Statements that serve as captions for the entire table do not need footnote letters.

aGaussian units are the same as cg emu for magnetostatics; Mx = maxwell, G = gauss, Oe = oersted; Wb = weber, V = volt, s = second, T = tesla, m = meter, A = ampere, J = joule, kg = kilogram, H = henry.

1. MULTIPART FIGURES

Figures compiled of more than one sub-figure presented side-by-side, or stacked. If a multipart figure is made up of multiple figure types (one part is lineart, and another is grayscale or color) the figure should meet the stricter guidelines.

1. FILE FORMATS FOR GRAPHICS

Format and save your graphics using a suitable graphics processing program that will allow you to create the images as PostScript (PS), Encapsulated PostScript (.EPS), Tagged Image File Format (.TIFF), Portable Document Format (.PDF), or Portable Network Graphics (.PNG) sizes them, and adjusts the resolution settings. If you created your source files in one of the following programs you will be able to submit the graphics without converting to a PS, EPS, TIFF, PDF, or PNG file: Microsoft Word, Microsoft PowerPoint, or Microsoft Excel. Though it is not required, it is strongly recommended that these files be saved in PDF format rather than DOC, XLS, or PPT. Doing so will protect your figures from common font and arrow stroke issues that occur when working on the files across multiple platforms. When submitting your final paper, your graphics should all be submitted individually in one of these formats along with the manuscript.

1. SIZING OF GRAPHICS

Most charts, graphs, and tables are one column wide (3.5 inches / 88 millimeters / 21 picas) or page wide (7.16 inches / 181 millimeters / 43 picas). The maximum depth a graphic can be is 8.5 inches (216 millimeters / 54 picas). When choosing the depth of a graphic, please allow space for a caption. Figures can be sized between column and page widths if the author chooses, however it is recommended that figures are not sized less than column width unless when necessary.

There is currently one publication with column measurements that do not coincide with those listed above. Proceedings of the IEEE has a column measurement of 3.25 inches (82.5 millimeters / 19.5 picas).

The final printed size of author photographs is exactly   
1 inch wide by 1.25 inches tall (25.4 millimeters x 31.75 millimeters / 6 picas x 7.5 picas). Author photos printed in editorials measure 1.59 inches wide by 2 inches tall (40 millimeters x 50 millimeters / 9.5 picas x 12 picas).

1. RESOLUTION

The proper resolution of your figures will depend on the type of figure it is as defined in the “Types of Figures” section. Author photographs, color, and grayscale figures should be at least 300dpi. Line art, including tables should be a minimum of 600dpi.

1. VECTOR ART

In order to preserve the figures’ integrity across multiple computer platforms, we accept files in the following formats: .EPS/.PDF/.PS. All fonts must be embedded or text converted to outlines in order to achieve the best-quality results.

1. COLOR SPACE

The term color space refers to the entire sum of colors that can be represented within the said medium. For our purposes, the three main color spaces are Grayscale, RGB (red/green/blue) and CMYK (cyan/magenta/yellow/black). RGB is generally used with on-screen graphics, whereas CMYK is used for printing purposes.

All color figures should be generated in RGB or CMYK color space. Grayscale images should be submitted in Grayscale color space. Line art may be provided in grayscale OR bitmap colorspace. Note that “bitmap colorspace” and “bitmap file format” are not the same thing. When bitmap color space is selected, .TIF/.TIFF/.PNG are the recommended file formats.

1. ACCEPTED FONTS WITHIN FIGURES

When preparing your graphics IEEE suggests that you use of one of the following Open Type fonts: Times New Roman, Helvetica, Arial, Cambria, and Symbol. If you are supplying EPS, PS, or PDF files all fonts must be embedded. Some fonts may only be native to your operating system; without the fonts embedded, parts of the graphic may be distorted or missing.

A safe option when finalizing your figures is to strip out the fonts before you save the files, creating “outline” type. This converts fonts to artwork what will appear uniformly on any screen.

1. USING LABELS WITHIN FIGURES
2. Figure Axis labels

Figure axis labels are often a source of confusion. Use words rather than symbols. As an example, write the quantity “Magnetization,” or “Magnetization M,” not just “M.” Put units in parentheses. Do not label axes only with units. As in Fig. 1, for example, write “Magnetization (A/m)” or “Magnetization (Am−1),” not just “A/m.” Do not label axes with a ratio of quantities and units. For example, write “Temperature (K),” not “Temperature/K.”

Multipliers can be especially confusing. Write “Magnetization (kA/m)” or “Magnetization (103 A/m).” Do not write “Magnetization (A/m) × 1000” because the reader would not know whether the top axis label in Fig. 1 meant 16000 A/m or 0.016 A/m. Figure labels should be legible, approximately 8 to 10 point type.

1. Subfigure Labels in Multipart Figures and Tables

Multipart figures should be combined and labeled before final submission. Labels should appear centered below each subfigure in 8 point Times New Roman font in the format of (a) (b) (c).

1. FILE NAMING

Figures (line artwork or photographs) should be named starting with the first 5 letters of the author’s last name. The next characters in the filename should be the number that represents the sequential location of this image in your article. For example, in author “Anderson’s” paper, the first three figures would be named ander1.tif, ander2.tif, and ander3.ps.

Tables should contain only the body of the table (not the caption) and should be named similarly to figures, except that ‘.t’ is inserted in-between the author’s name and the table number. For example, author Anderson’s first three tables would be named ander.t1.tif, ander.t2.ps, ander.t3.eps.

Author photographs should be named using the first five characters of the pictured author’s last name. For example, four author photographs for a paper may be named: oppen.ps, moshc.tif, chen.eps, and duran.pdf.

If two authors or more have the same last name, their first initial(s) can be substituted for the fifth, fourth, third... letters of their surname until the degree where there is differentiation. For example, two authors Michael and Monica Oppenheimer’s photos would be named oppmi.tif, and oppmo.eps.

1. REFERENCING A FIGURE OR TABLE WITHIN YOUR PAPER

When referencing your figures and tables within your paper, use the abbreviation “Fig.” even at the beginning of a sentence. Do not abbreviate “Table.” Tables should be numbered with Roman Numerals.

1. CHECKING YOUR FIGURES: THE IEEE GRAPHICS ANALYZER

The IEEE Graphics Analyzer enables authors to pre-screen their graphics for compliance with IEEE Access standards before submission. The online tool, located at <http://graphicsqc.ieee.org/>, allows authors to upload their graphics in order to check that each file is the correct file format, resolution, size and colorspace; that no fonts are missing or corrupt; that figures are not compiled in layers or have transparency, and that they are named according to the IEEE Access naming convention. At the end of this automated process, authors are provided with a detailed report on each graphic within the web applet, as well as by email.

For more information on using the Graphics Analyzer   
or any other graphics related topic, contact the IEEE Graphics Help Desk by e-mail at [graphics@ieee.org](mailto:graphics@ieee.org).

1. SUBMITTING YOUR GRAPHICS

Because IEEE will do the final formatting of your paper,

you do not need to position figures and tables at the top and bottom of each column. In fact, all figures, figure captions, and tables can be placed at the end of your paper. In addition to, or even in lieu of submitting figures within your final manuscript, figures should be submitted individually, separate from the manuscript in one of the file formats listed above in section VI-J. Place figure captions below the figures; place table titles above the tables. Please do not include captions as part of the figures, or put them in “text boxes” linked to the figures. Also, do not place borders around the outside of your figures.

1. COLOR PROCESSING / PRINTING IN IEEE JOURNALS

All IEEE Transactions, Journals, and Letters allow an author to publish color figures on IEEE Xplore® at no charge, and automatically convert them to grayscale for print versions. In most journals, figures and tables may alternatively be printed in color if an author chooses to do so. Please note that this service comes at an extra expense to the author. If you intend to have print color graphics, include a note with your final paper indicating which figures or tables you would like to be handled that way, and stating that you are willing to pay the additional fee.

CONCLUSION

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.

APPENDIX

Appendixes, if needed, appear before the acknowledgment.

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank ... .” Instead, write “F. A. Author thanks ... .” In most cases, sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page, not here.

REFERENCES AND FOOTNOTES

1. REFERENCES

References need not be cited in text. When they are, they appear on the line, in square brackets, inside the punctuation. Multiple references are each numbered with separate brackets. When citing a section in a book, please give the relevant page numbers. In text, refer simply to the reference number. Do not use “Ref.” or “reference” except at the beginning of a sentence: “Reference [3] shows ... .” Please do not use automatic endnotes in Word, rather, type the reference list at the end of the paper using the “References” style.

Reference numbers are set flush left and form a column of their own, hanging out beyond the body of the reference. The reference numbers are on the line, enclosed in square brackets. In all references, the given name of the author or editor is abbreviated to the initial only and precedes the last name. Use them all; use et al. only if names are not given. Use commas around Jr., Sr., and III in names. Abbreviate conference titles. When citing IEEE transactions, provide the issue number, page range, volume number, year, and/or month if available. When referencing a patent, provide the day and the month of issue, or application. References may not include all information; please obtain and include relevant information. Do not combine references. There must be only one reference with each number. If there is a URL included with the print reference, it can be included at the end of the reference.

Other than books, capitalize only the first word in a paper title, except for proper nouns and element symbols. For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation See the end of this document for formats and examples of common references. For a complete discussion of references and their formats, see the IEEE style manual at www.ieee.org/authortools.

1. FOOTNOTES

Number footnotes separately in superscripts (Insert| Footnote).[[2]](#footnote-2) Place the actual footnote at the bottom of the column in which it is cited; do not put footnotes in the reference list (endnotes). Use letters for table footnotes (see Table I).

SUBMITTING YOUR PAPER FOR REVIEW

1. REVIEW STAGE USING WORD 6.0 OR HIGHER

If you want to submit your file with one column electronically, please do the following:

--First, click on the View menu and choose Print Layout.

--Second, place your cursor in the first paragraph. Go to the Format menu, choose Columns, choose one column Layout, and choose “apply to whole document” from the dropdown menu.

--Third, click and drag the right margin bar to just over 4 inches in width.

The graphics will stay in the “second” column, but you can drag them to the first column. Make the graphic wider to push out any text that may try to fill in next to the graphic.

1. FINAL STAGE USING WORD 6.0

When you submit your final version (after your paper has been accepted), print it in two-column format, including figures and tables. You must also send your final manuscript on a disk, via e-mail, or through a Web manuscript submission system as directed by the society contact. You may use Zip for large files, or compress files using Compress, Pkzip, Stuffit, or Gzip.

Also, send a sheet of paper or PDF with complete contact information for all authors. Include full mailing addresses, telephone numbers, fax numbers, and e-mail addresses. This information will be used to send each author a complimentary copy of the journal in which the paper appears. In addition, designate one author as the “corresponding author.” This is the author to whom proofs of the paper will be sent. Proofs are sent to the corresponding author only.

1. REVIEW STAGE USING SCHOLARONE® MANUSCRIPTS

Contributions to the Transactions, Journals, and Letters may be submitted electronically on IEEE’s on-line manuscript submission and peer-review system, ScholarOne® Manuscripts. You can get a listing of the publications that participate in ScholarOneat http://www.ieee.org/  
publications\_standards/publications/authors/authors\_submission.html First check if you have an existing account. If there is none, please create a new account. After logging in, go to your Author Center and click “Submit First Draft of a New Manuscript.”

Along with other information, you will be asked to select the subject from a pull-down list. Depending on the journal, there are various steps to the submission process; you must complete all steps for a complete submission. At the end of each step you must click “Save and Continue”; just uploading the paper is not sufficient. After the last step, you should see a confirmation that the submission is complete. You should also receive an e-mail confirmation. For inquiries regarding the submission of your paper on ScholarOne Manuscripts, please contact oprs-support@ieee.org or call +1 732 465 5861.

ScholarOne Manuscripts will accept files for review in various formats. Please check the guidelines of the specific journal for which you plan to submit.

You will be asked to file an electronic copyright form immediately upon completing the submission process (authors are responsible for obtaining any security clearances). Failure to submit the electronic copyright could result in publishing delays later. You will also have the opportunity to designate your article as “open access” if you agree to pay the IEEE open access fee.

1. FINAL STAGE USING SCHOLARONE MANUSCRIPTS

Upon acceptance, you will receive an email with specific instructions regarding the submission of your final files. To avoid any delays in publication, please be sure to follow these instructions. Most journals require that final submissions be uploaded through ScholarOne Manuscripts, although some may still accept final submissions via email. Final submissions should include source files of your accepted manuscript, high quality graphic files, and a formatted pdf file. If you have any questions regarding the final submission process, please contact the administrative contact for the journal.

In addition to this, upload a file with complete contact information for all authors. Include full mailing addresses, telephone numbers, fax numbers, and e-mail addresses. Designate the author who submitted the manuscript on ScholarOne Manuscripts as the “corresponding author.” This is the only author to whom proofs of the paper will be sent.

1. COPYRIGHT FORM

Authors must submit an electronic IEEE Copyright Form (eCF) upon submitting their final manuscript files. You can access the eCF system through your manuscript submission system or through the Author Gateway. You are responsible for obtaining any necessary approvals and/or security clearances. For additional information on intellectual property rights, visit the IEEE Intellectual Property Rights department web page at http://www.ieee.org/publications\_  
standards/publications/rights/index.html.

IEEE PUBLISHING POLICY

The general IEEE policy requires that authors should only submit original work that has neither appeared elsewhere for publication, nor is under review for another refereed publication. The submitting author must disclose all prior publication(s) and current submissions when submitting a manuscript. Do not publish “preliminary” data or results. The submitting author is responsible for obtaining agreement of all coauthors and any consent required from employers or sponsors before submitting an article. The IEEE Access

Department strongly discourages courtesy authorship; it is the obligation of the authors to cite only relevant prior work.

The IEEE Access Department does not publish conference records or proceedings, but can publish articles related to conferences that have undergone rigorous peer review. Minimally, two reviews are required for every article submitted for peer review.

PUBLICATION PRINCIPLES

The two types of contents of that are published are; 1) peer-reviewed and 2) archival. The Transactions and Journals Department publishes scholarly articles of archival value as well as tutorial expositions and critical reviews of classical subjects and topics of current interest.

Authors should consider the following points:

1. Technical papers submitted for publication must advance the state of knowledge and must cite relevant prior work.
2. The length of a submitted paper should be commensurate with the importance, or appropriate to the complexity, of the work. For example, an obvious extension of previously published work might not be appropriate for publication or might be adequately treated in just a few pages.
3. Authors must convince both peer reviewers and the editors of the scientific and technical merit of a paper; the standards of proof are higher when extraordinary or unexpected results are reported.
4. Because replication is required for scientific progress, papers submitted for publication must provide sufficient information to allow readers to perform similar experiments or calculations and use the reported results. Although not everything need be disclosed, a paper must contain new, useable, and fully described information. For example, a specimen’s chemical composition need not be reported if the main purpose of a paper is to introduce a new measurement technique. Authors should expect to be challenged by reviewers if the results are not supported by adequate data and critical details.
5. Papers that describe ongoing work or announce the latest technical achievement, which are suitable for presentation at a professional conference, may not be appropriate for publication.

REFERENCES

[1] J. J. Estruch, N. B. Carozzi, N. Desai, N. B. Duck, G. W. Warren, and M. G. Koziel, ‘Transgenic plants: an emerging approach to pest control’, *Nat Biotechnol*, vol. 2, no. 15, pp. 137–141, 1997, doi: https://doi.org/10.1038/nbt0297-137.

[2] V. Stejskal, T. Vendl, Z. Li, and R. Aulicky, ‘Efficacy of visual evaluation of insect-damaged kernels of malting barley by Sitophilus granarius from various observation perspectives’, *J Stored Prod Res*, vol. 89, p. 101711, 2020, doi: https://doi.org/10.1016/j.jspr.2020.101711.

[3] Y. Sun, X. Liu, M. Yuan, L. Ren, J. Wang, and Z. Chen, ‘Automatic in-trap pest detection using deep learning for pheromone-based Dendroctonus valens monitoring’, *Biosyst Eng*, vol. 176, pp. 140–150, 2018, doi: https://doi.org/10.1016/j.biosystemseng.2018.10.012.

[4] S. Cui, P. Ling, H. Zhu, and H. M. Keener, ‘Plant pest detection using an artificial nose system: A review’, Feb. 01, 2018, *MDPI AG*. doi: 10.3390/s18020378.

[5] W. Li, T. Zheng, Z. Yang, M. Li, C. Sun, and X. Yang, ‘Classification and detection of insects from field images using deep learning for smart pest management: A systematic review’, *Ecol Inform*, vol. 66, p. 101460, 2021, doi: https://doi.org/10.1016/j.ecoinf.2021.101460.

[6] J. Wei *et al.*, ‘Improving the Accuracy of Agricultural Pest Identification: Application of AEC-YOLOv8n to Large-Scale Pest Datasets’, *Agronomy*, vol. 14, no. 8, Aug. 2024, doi: 10.3390/agronomy14081640.

[7] X. Zhao, W. Li, Y. Zhang, T. A. Gulliver, S. Chang, and Z. Feng, ‘A Faster RCNN-Based Pedestrian Detection System’, in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, 2016, pp. 1–5. doi: 10.1109/VTCFall.2016.7880852.

[8] F. Ali, H. Qayyum, and M. J. Iqbal, ‘Faster-PestNet: A Lightweight Deep Learning Framework for Crop Pest Detection and Classification’, *IEEE Access*, vol. 11, pp. 104016–104027, 2023, doi: 10.1109/ACCESS.2023.3317506.

[9] X. Wu, C. Zhan, Y.-K. Lai, M.-M. Cheng, and J. Yang, ‘IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition’, Long Beach, CA, USA, Jan. 2020. doi: 10.1109/CVPR.2019.00899.

[10] D. G. Lowe, ‘Object recognition from local scale-invariant features’, in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999, pp. 1150–1157 vol.2. doi: 10.1109/ICCV.1999.790410.

[11] N. Dalal and B. Triggs, ‘Histograms of oriented gradients for human detection’, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, 2005, pp. 886–893 vol. 1. doi: 10.1109/CVPR.2005.177.

[12] S. Shan, P. Yang, X. Chen, and W. Gao, ‘AdaBoost Gabor Fisher Classifier for Face Recognition’, in *Analysis and Modelling of Faces and Gestures*, W. Zhao, S. Gong, and X. Tang, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 279–292.

[13] L. Du, R. Zhang, and X. Wang, ‘Overview of two-stage object detection algorithms’, in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2020. doi: 10.1088/1742-6596/1544/1/012033.

[14] R. Girshick, J. Donahue, T. Darrell, and J. Malik, ‘Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation’, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587. doi: 10.1109/CVPR.2014.81.

[15] R. Girshick, ‘Fast R-CNN’, in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448. doi: 10.1109/ICCV.2015.169.

[16] 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, 2017, doi: 10.1109/TPAMI.2016.2577031.

[17] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, ‘A survey of modern deep learning based object detection models’, *Digit Signal Process*, vol. 126, p. 103514, 2022, doi: https://doi.org/10.1016/j.dsp.2022.103514.

[18] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, ‘Focal Loss for Dense Object Detection’, *IEEE Trans Pattern Anal Mach Intell*, vol. 42, no. 2, pp. 318–327, 2020, doi: 10.1109/TPAMI.2018.2858826.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ‘You Only Look Once: Unified, Real-Time Object Detection’, Jun. 2015, [Online]. Available: http://arxiv.org/abs/1506.02640

[20] M. A. R. Alif and M. Hussain, ‘YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain’, Jun. 2024, [Online]. Available: http://arxiv.org/abs/2406.10139

[21] J. Redmon and A. Farhadi, ‘YOLO9000: Better, Faster, Stronger’, Dec. 2016, [Online]. Available: http://arxiv.org/abs/1612.08242

[22] J. Redmon and A. Farhadi, ‘YOLOv3: An Incremental Improvement’, Apr. 2018, [Online]. Available: http://arxiv.org/abs/1804.02767

[23] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, ‘YOLOv4: Optimal Speed and Accuracy of Object Detection’, Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.10934

[24] K. He, X. Zhang, S. Ren, and J. Sun, ‘Deep Residual Learning for Image Recognition’. [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[25] Z. Ma, M. Li, and Y. Wang, ‘PAN: Path Integral Based Convolution for Deep Graph Neural Networks’, Apr. 2019, [Online]. Available: http://arxiv.org/abs/1904.10996

[26] Z. Yao, Y. Cao, S. Zheng, G. Huang, and S. Lin, ‘Cross-Iteration Batch Normalization’. [Online]. Available: https://aka.ms/cbn.

[27] S. He *et al.*, ‘Computer-Vision Benchmark Segment-Anything Model (SAM) in Medical Images: Accuracy in 12 Datasets’, Apr. 2023, [Online]. Available: http://arxiv.org/abs/2304.09324

[28] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, ‘YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors’, in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 7464–7475. doi: 10.1109/CVPR52729.2023.00721.

[29] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, ‘YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information’, Feb. 2024, [Online]. Available: http://arxiv.org/abs/2402.13616

[30] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, ‘End-to-End Object Detection with Transformers’, in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds., Cham: Springer International Publishing, 2020, pp. 213–229.

[31] A. Sharma, V. Kumar, and L. Longchamps, ‘Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species’, *Smart Agricultural Technology*, p. 100648, Nov. 2024, doi: 10.1016/j.atech.2024.100648.

[32] D. Sun *et al.*, ‘Efficient Tobacco Pest Detection in Complex Environments Using an Enhanced YOLOv8 Model’, *Agriculture (Switzerland)*, vol. 14, no. 3, Mar. 2024, doi: 10.3390/agriculture14030353.

[33] L. Shen, B. Lang, and Z. Song, ‘DS-YOLOv8-Based Object Detection Method for Remote Sensing Images’, *IEEE Access*, vol. 11, pp. 125122–125137, 2023, doi: 10.1109/ACCESS.2023.3330844.

[34] J. M. Johnson and T. M. Khoshgoftaar, ‘Survey on deep learning with class imbalance’, *J Big Data*, vol. 6, no. 1, Dec. 2019, doi: 10.1186/s40537-019-0192-5.

[35] J. N. OPARA, R. MORIWAKI, and P. CHUN, ‘Delineating Landslide and Debris Flow Detection in Japan through Aerial Photography: A YOLO v8 Approach to Disaster Management’, *Intelligence, Informatics and Infrastructure*, vol. 5, no. 1, pp. 111–123, 2024, doi: 10.11532/jsceiiai.5.1\_111.

[36] A. Vaswani *et al.*, ‘Attention is all you need’, in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, in NIPS’17. Red Hook, NY, USA: Curran Associates Inc., 2017, pp. 6000–6010.

[37] A. Wang *et al.*, ‘YOLOv10: Real-Time End-to-End Object Detection’, May 2024, [Online]. Available: http://arxiv.org/abs/2405.14458

[38] G. Jocher, A. Chaurasia, and J. Qiu, ‘YOLO11 by Ultralytics’, Oct. 2024. [Online]. Available: https://github.com/ultralytics/ultralytics

a place and/or date of birth (list place, then date). Next, the author’s educational background is listed. The degrees should be listed with type of degree in what field, which institution, city, state, and country, and year the degree was earned. The author’s major field of study should be lower-cased.

The second paragraph uses the pronoun of the person (he or she) and not the author’s last name. It lists military and work experience, including summer and fellowship jobs. Job titles are capitalized. The current job must have a location; previous positions may be listed without one. Information concerning previous publications may be included. Try not to list more than three books or published articles. The format for listing publishers of a book within the biography is: title of book (publisher name, year) similar to a reference. Current and previous research interests end the paragraph.

The third paragraph begins with the author’s title and last name (e.g., Dr. Smith, Prof. Jones, Mr. Kajor, Ms. Hunter). List any memberships in professional societies other than the IEEE. Finally, list any awards and work for IEEE committees and publications. If a photograph is provided, it should be of good quality, and professional-looking. Following are two examples of an author’s biography.

**SECOND B. AUTHOR** was born in Greenwich Village, New York, NY, USA in 1977. He received the B.S. and M.S. degrees in aerospace engineering from the University of Virginia, Charlottesville, in 2001 and the Ph.D. degree in mechanical engineering from Drexel University, Philadelphia, PA, in 2008.

From 2001 to 2004, he was a Research Assistant with the Princeton Plasma Physics Laboratory. Since 2009, he has been an Assistant Professor with the Mechanical Engineering Department, Texas A&M University, College Station. He is the author of three books, more than 150 articles, and more than 70 inventions. His research interests include high-pressure and high-density nonthermal plasma discharge processes and applications, microscale plasma discharges, discharges in liquids, spectroscopic diagnostics, plasma propulsion, and innovation plasma applications. He is an Associate Editor of the journal *Earth*, *Moon*, *Planets*, and holds two patents.

Dr. Author was a recipient of the International Association of Geomagnetism and Aeronomy Young Scientist Award for Excellence in 2008, and the IEEE Electromagnetic Compatibility Society Best Symposium Paper Award in 2011.

**THIRD C. AUTHOR, JR.** (M’87) received the B.S. degree in mechanical engineering from National Chung Cheng University, Chiayi, Taiwan, in 2004 and the M.S. degree in mechanical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2006. He is currently pursuing the Ph.D. degree in mechanical engineering at Texas A&M University, College Station, TX, USA.

From 2008 to 2009, he was a Research Assistant with the Institute of Physics, Academia Sinica, Tapei, Taiwan. His research interest includes the development of surface processing and biological/medical treatment techniques using nonthermal atmospheric pressure plasmas, fundamental study of plasma sources, and fabrication of micro- or nanostructured surfaces.

Mr. Author’s awards and honors include the Frew Fellowship (Australian Academy of Science), the I. I. Rabi Prize (APS), the European Frequency and Time Forum Award, the Carl Zeiss Research Award, the William F. Meggers Award and the Adolph Lomb Medal (OSA).

1. 1125 layer - 2hours/epoch [↑](#footnote-ref-1)
2. It is recommended that footnotes be avoided (except for the unnumbered footnote with the receipt date on the first page). Instead, try to integrate the footnote information into the text. [↑](#footnote-ref-2)