LMS-YOLO11n: A Lightweight Multi-Scale Weed Detection Model

YaJun Zhang[1], Yu Xu[2], Jie Hou[3], YanHai Song[4],
Department School of Software, Xinjiang University, Urumqi, China[1,2,4]
Guangzhou Xinhe Information Technology Corporation, Guangzhou, China[3]

***Abstract*—With the advancement of precision agriculture, efficient and accurate weed detection has emerged as a pivotal task in modern crop management. Current weed detection methods face dual challenges: inadequate extraction of detailed features and edge information, coupled with the necessity for real-time performance. To address these issues, this paper proposes a lightweight multi-scale weed detection model based on YOLOv11n (You-only-look-once-11). Our approach incorporates three innovative components: (1) A fast-gated lightweight unit combined with C3K2 to enhance local and global interaction capabilities of weed features. (2) An adaptive hierarchical feature fusion network based on HSFPN, which improves the extraction of weed edge information. (3) A lightweight group convolution detection head module that captures multi-scale feature details while maintaining a lightweight structure. Experimental validation on two public datasets, CottonWeedDet3 and CottonWeed2, demonstrates that our model achieves an mAP50 improvement of 2.5% on CottonWeedDet3 and 1.9% on CottonWeed2 compared to YOLOv11n, with a 37% reduction in parameters and a 26% decrease in computational effort.**

Keywords*—*You-only-look-once-11; weed; lightweight; group convolution

I. Introduction

Modern agriculture faces numerous challenges that hinder productivity and sustainable development. Weeds are a major threat, directly impacting crop yield and food security. Weeds compete with crops for light, water, and soil nutrients, spreading diseases and pests, significantly reducing crop yield, and causing economic losses [1-2]. Selective herbicides and manual weeding are the two major weed management techniques used today; the former entails evenly applying herbicides throughout fields. This method results in significant waste, as most herbicides are sprayed on crops or bare soil, rather than directly on the weeds. Additionally, excessive herbicide use harms the ecosystem. Manual weeding, on the other hand, is costly and difficult to scale for large-scale agricultural operations.

With the development of AI technology, precision agriculture offers a solution to these problems [3-4], with the key first step being the accurate and rapid detection of weed locations [5]. Therefore, in-depth research on weed detection technology is crucial for the development of precision agriculture, contributing to the future efficiency, precision, and sustainability of farming.

Early weed detection methods were mostly based on machine learning, such as Kumar and Prema's [6] Wrapped Curve Transform Angle Texture Pattern extraction method, which improved weed identification accuracy in fields. Sujaritha et al. [7] proposed a circular leaf pattern extraction method based on morphological operations, combined with rotational invariance and wavelet decomposition, enabling automatic weed and crop recognition and efficient removal in sugarcane fields. However, these methods struggle to handle challenges such as the complexity of field environments, weed species diversity, and lighting changes, resulting in poor detection performance and instability, which limits their application in diverse environments. In contrast, deep learning uses convolutional neural networks to extract both global and local features of weeds, compensating for the shortcomings of machine learning in feature extraction. As a result, deep learning-based weed detection has become the mainstream method.

While deep learning-based methods outperform machine learning in terms of accuracy, they struggle to meet the lightweight requirements of edge devices, making weed detection on edge devices a new direction in object detection. This study faces challenges such as high similarity between weed species, occlusion issues affecting detection accuracy, and the deployment limitations of edge devices. To address these challenges, this paper proposes the LMS-YOLO11n weed detection method based on YOLO11n. The main contributions of the LMS-YOLO11n model are as follows:

1) To meet the demands of detail extraction and real-time performance in weed detection, this paper proposes the lightweight multi-scale feature extraction module FastGLU, combined with CGLU's convolutional gating mechanism and FasterNet's lightweight characteristics. It extracts key channel information through partial convolution (PConv) and uses CGLU to enhance the interaction between local and global features, reducing computational costs while achieving efficient and diverse feature extraction.
2) To address the challenge of weed edge information extraction, this paper designs the adaptive hierarchical feature fusion network (AHFPN). By combining the ideas of HSFPN and PAN, the feature fusion mechanism is improved to enhance sensitivity and capability in edge information extraction, optimize the interaction and fusion of multi-scale features, and improve adaptability to weed diversity and various growth stages, while reducing computational burden.
3) To meet the real-time requirements of weed detection, this paper introduces the lightweight group convolution detection head (LGCD) module. By incorporating group convolution into the position regression branch, the computational load and parameter count are significantly reduced, and kernel size optimization improves the ability

to capture multi-level feature details, balancing feature extraction richness with model efficiency to meet the deployment requirements of edge devices.

This paper has the following structure: Section II examines the state of domestic and international weed detection research; Section III provides further details about the LMS-YOLO11n.approach; experiments in Section IV confirm the model's generalization performance; and a summary of the work and recommendations for future research are provided in Section V.

II. Related Work

Object detection methods can be broadly categorized into single-stage and two-stage models. Two-stage detection models generate candidate regions quickly and refine them in a second processing stage. Typical models include RCNN [8] and Fast-RCNN [9]. Zhang [10] and colleagues successfully detected weeds and soybeans in complex backdrops by optimizing the Faster R-CNN method with VGG19-CBAM as the backbone network, achieving successful detection of soybeans and weeds in complex backgrounds. Ozcan et al. [11] compared the performance of single-stage and two-stage CNN models in precision agriculture and found that Faster R-CNN Inception v2 offers higher accuracy. However, when training and inference time are critical, the SSD MobileNet v2 model significantly improves accuracy with increased training data. Li et al. [12] proposed an improved Faster R-CNN model for automatic detection of hydroponic lettuce seedlings, achieving an accuracy of 86.2% through enhancement techniques, outperforming models like RetinaNet, SSD, Cascade RCNN, and FCOS. Although two-stage detection methods are generally more accurate than single-stage methods, they require significant computational resources, making them difficult to deploy on mobile devices. Moreover, the longer detection time limits their ability to meet detection in real-time requirements.

Due to the limitations of two-stage detection, there is growing interest in single-stage detection methods, exemplified by SSD [13] and YOLO [14-18]. Unlike two-stage methods, single-stage detection integrates region proposal, classification, and regression into a single network, significantly improving speed and efficiency. Chen et al. [19] proposed the YOLO-sesame model, an improved YOLOv4 variant that incorporates Local Importance Pooling (LIP) and SE modules to enhance feature extraction. The model also uses an Adaptive Spatial Feature Fusion (ASFF) structure to optimize the detection of objects of varying sizes, improving both real-time performance and accuracy in sesame field weed detection. Hong et al. [20] presented an enhanced YOLOv5 algorithm for effective asparagus identification in intricate settings. The model incorporates Coordinate Attention (CA) in the backbone network to emphasize growth features of asparagus and replaces PANet with BiFPN to enhance feature propagation and reuse, significantly improving support for intelligent mechanical harvesting under various weather conditions. A network of convolutional neuron models called RIC-Net, which combines residual structures with Inception, was proposed by Zhao et al. [21]. The model replaces MLP layers with 1D convolutions for optimized feature detection and integrates CBAM modules with weighted operations to highlight diseased areas, improving classification accuracy for leaf diseases in maize, potatoes, and tomatoes. Song et al. [22] developed an improved YOLOv5 algorithm by replacing the backbone with MobileNetv2 to reduce model complexity. ECANet attention mechanisms were introduced to enhance detailed feature extraction for soybean leaves, and CIOU_Loss + DIOU_NMS was used to improve accuracy and robustness, particularly for dense occlusion and small object detection in precision agriculture spraying. Zhang et al. [23] proposed CCCS-YOLO, an improved YOLOv5-based algorithm. The model integrates Faster_Block into YOLOv5s's C3 module to create C3_Faster, simplifying the network structure and enhancing detection. It improves the convolutional block in the head for better target-background differentiation, replaces the neck's upsampling module with the lightweight CARAFE module for small object detection and contextual information fusion, and uses Soft-NMS-EIoU to enhance detection accuracy in dense scenarios. Guo et al. [24] proposed LW-YOLOv8n, a lightweight weed detection model. The model integrates SERMAttention with SE and SRM modules to capture global information, incorporates lightweight Context Guided Blocks in C2f layers to enhance local and contextual feature learning, and introduces an improved BiFPN network in the neck for weighted multi-scale feature fusion. This method is appropriate for edge devices with limited resources as it lowers parameters and complexity while preserving excellent detection accuracy. Fan et al. [25] presented YOLO-WDNet, a model for lightweight weed identification. It replaces CSP-Darknet53 with ShuffleNet v2 as the backbone to reduce parameters and complexity, designs a Parallel Hybrid Attention Mechanism (PHAM) to focus on regions of interest, improves BiFPN in the neck for multi-scale and overlapping plant feature recognition, and proposes an EIOU loss function to enhance detection accuracy in dense scenarios.

Despite significant advancements, a gap persists in the development of single-stage detection models that combine high accuracy with sufficient lightness and efficiency for deployment on resource-limited edge devices. The work presented in this paper endeavors to address this gap by introducing a novel, lightweight single-stage detection model. This model integrates advanced convolutional gating mechanisms, optimized feature fusion strategies, and a streamlined detection head leveraging grouped convolutions, all tailored to elevate detection accuracy and efficiency specifically within agricultural applications.

III. Methodology

A. YOLO11 Principle

YOLO11, the latest model in the YOLO series, was released by Ultralytics in 2024. Based on structural complexity and size, YOLO11 is available in five versions: YOLO11n, YOLO11s, YOLO11m, YOLO11l, and YOLO11x.YOLO11n, the version with the smallest computational and parameter requirements, is designed for weed detection scenarios that demand real-time performance and limited computational resources. Therefore, this study selects YOLO11n as the baseline model.YOLO11n consists of three main components: Backbone, Neck, and Head. The Backbone replaces the C2f module from YOLOv8 with the latest C3K2 module, significantly improving feature extraction efficiency. The Neck continues to use the FPN+PAN structure for feature fusion, while the Head incorporates depthwise separable convolutions, significantly reducing computation and parameter requirements. Although

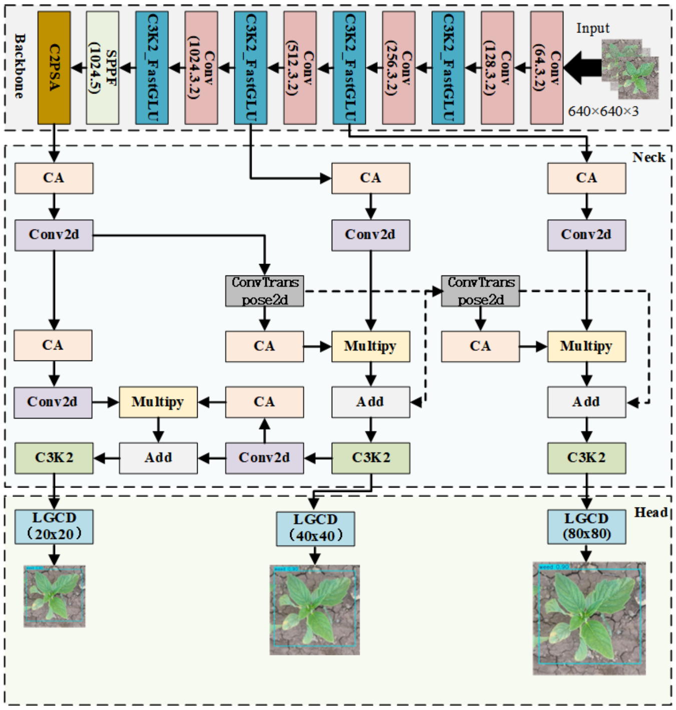

Fig. 1. Structure of LMS-YOLO11n.

YOLO11n is the most lightweight version, its computational complexity remains 6.3GFLOPs, and its parameter size is 2.58MB, which still poses challenges for real-time detection and edge computing deployment.

B. Lightweight Multi-Scale Weed Detection Model

In deep learning-based weed detection tasks, the scale and complexity of the model directly determine its practical effectiveness. Although YOLO11 surpasses many mainstream object detection models in speed, it contains significant redundant features. These redundant features are primarily generated by convolutional computations in the backbone network, consuming substantial computational resources, increasing complexity, and reducing inference speed. Additionally, the similar textures of crop seedlings and weeds, coupled with multi-scale features, make YOLO11 less effective at extracting features under complex lighting conditions. To address the need for lightweight models and real-time detection, while enhancing the extraction of fine-grained weed features and edge information, this study proposes the LMS-YOLO11n model based on YOLO11n. LMS-YOLO11n integrates the C3K2_FCGLU module into the YOLO11n framework to replace the original C3K2 module, enabling more efficient weed feature extraction. Furthermore, by introducing the AHFPN designed based on HSFPN [26], the neck network is optimized to improve the recognition and fusion of multi-scale overlapping plant features. Finally, the LGCD module, based on grouped convolution [27], is used to refine the Head, enhancing multi-scale information capture while reducing parameters and computation. Fig. 1 displays the LMS-YOLO11n structure with an input picture size of 640 x 640 x 3.

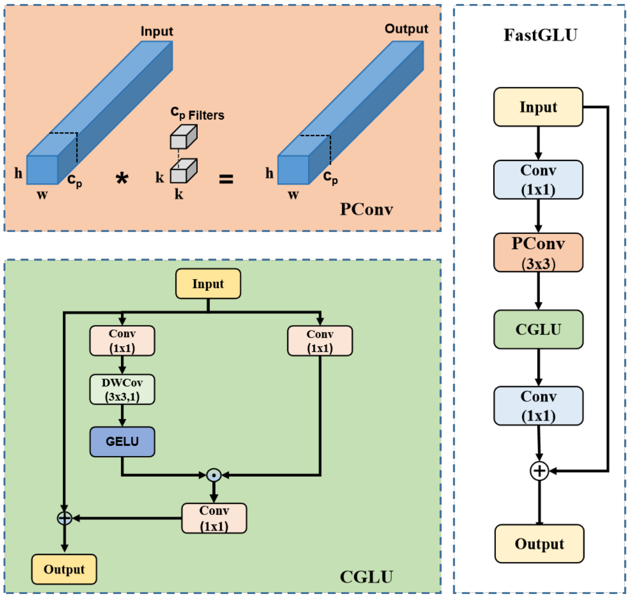

Fig. 2. Structure of CGLU, PConv and FastGLU.

C. C2f_FastGLU

To address the requirements for detail extraction and real-time processing in weed detection, this paper proposes a Fast Gated Lightweight Unit (FastGLU), as shown in Fig. 2. FastGLU captures fine-grained feature information, enhancing the model's ability to perceive image details, expand the receptive field, and extract local features. Additionally, it excels in optimizing multi-channel information usage, reducing parameters and computational costs, maintaining gradient flow, and enhancing spatial feature extraction. This allows the model to efficiently handle weeds of varying sizes and shapes. FasterNet [28] introduced the concept of Partial Convolution.PConv is a convolutional method designed to improve data processing efficiency and reduce memory overhead. It applies standard convolutions to a subset of input channels to effectively extract spatial features while omitting convolutions on other channels, thereby reducing computational and memory demands. Specifically, it selects the first and last consecutive c_p channels as representatives of the input feature map, assuming the input and output feature maps have the same number of channels. This design not only simplifies computation but also optimizes memory access efficiency, enabling effective feature representation. By applying convolutions only to a subset of input channels to extract spatial features while ignoring others, its computational complexity is defined in Eq. (1).

$$\begin{aligned} F_{Conv} &= h \times w \times k^2 \times c^2 \\ F_{PConv} &= h \times w \times k^2 \times c_p^2 \end{aligned} \tag{1}$$

In this equation, h and w represent the height and width of the feature map, k denotes the kernel size, c is the number of input feature map channels, and c_p represents the selected input channels used for spatial feature extraction in the PConv operation. In this study, c_p is set to 1/4 of c, reducing the computational cost of PConv to just 1/16 that of a standard convolution.

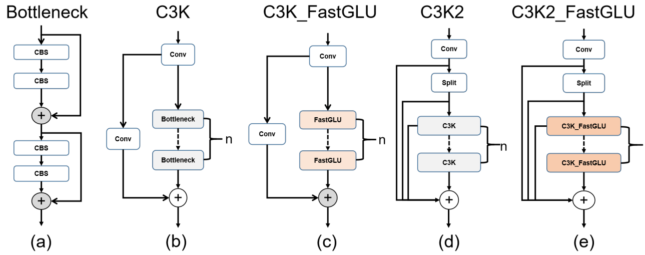

Fig. 3. (a) Bottleneck; (b) C3K; (c) C3K_FastGLU; (d) C3K2; (e) C3K2_FastGLU structure.

The Faster Block accelerates network processing by reducing computational and memory access demands. Its structure consists of a PConv layer followed by two pointwise convolution (PWConv) layers. However, the Faster Block has a limited receptive field, and PConv processes only part of the channel information, which hinders fine-grained weed feature extraction. The Gated Linear Unit (GLU) is an activation mechanism designed to enhance the extraction of complex features, initially used in language processing and sequence modeling tasks. Currently, GLU [29] has evolved into several variants, including the Gated Recurrent Unit, Depthwise Separable GLU, FFN with SE module, and Convolutional Gated Linear Unit [30]. In this study, CGLU is integrated into PConv to enhance fine-grained local feature extraction and optimize the interaction between local and global feature information. CGLU first employs two parallel 1x1 convolutions for per-channel control, with one feature map further processed by a 3x3 depthwise separable convolution to capture local features. These features are then fed into the Gated Linear Unit. A portion of the features is activated by the GELU function to serve as a gating signal, which multiplies with another feature set to enable channel attention control, enhancing feature selection and emphasis.

The primary structure of the C3K2 module in YOLO11 Fig. 3(d) is based on the C3K module. The C3K module Fig. 3(b) consists of standard convolutions and Bottleneck units. The Bottleneck unit Fig. (3a) is composed of CBS modules.CBS is a fundamental convolutional unit comprising convolution operations, batch normalization, and an activation function. The proposed C3K2_FastGLU module Fig. (3e) replaces the C3K module in C3K2 with C3K_FastGLU.The backbone network faces several bottleneck issues. Introducing C3K2_FastGLU effectively reduces computational cost, significantly improves multi-scale feature extraction, alleviates information transmission bottlenecks, and maintains efficient feature representation and generalization in lightweight designs. Therefore, in the YOLO11n backbone, the C3K2 modules in the P2, P3, P4, and P5 layers are replaced with C3K2_FastGLU.

D. AHFPN

To meet the deployment requirements of weed detection on edge devices, this study uses HSFPN to fuse extracted features. HSFPN consists of a Channel Attention (CA) module and a Semantic Feature Fusion (SFF) module, as illustrated in Fig. 4.

The Channel Attention (CA) module applies average pooling and max pooling to each channel's features, extracting the most relevant and average information for each channel. The pooled average and maximum results are combined, and the Sigmoid function calculates the weight for each channel. Finally, the weights are multiplied by the corresponding feature maps to filter redundant data. Additionally, a 1x1 convolution is used to adjust channel dimensions to 256, ensuring compatibility across different scales.

The Semantic Feature Fusion (SFF) module employs weights from higher-level characteristics to selectively integrate key semantic data derived from lower-level characteristics. The process includes: 1) applying a 3x3 transposed convolution with a stride of 2 to process higher-level features; 2) aligning the transposed higher-level features' dimensions with those of the lower-level features by the use of bilinear interpolation;3) employing the CA module to convert higher-level features into weights; and 4) combining the optimized lower-level features with higher-level characteristics to improve the depiction of aspects. The specific definition is given in Eq. (2):

$$\begin{aligned} f_{att} &= BL(T - Conv(f_{high})) \\ f_{out} &= f_{low} * CA(f_{att}) + f_{att} \end{aligned} \tag{2}$$

Include among these $f_{high} \in R^{C \times H \times W}$, $f_{low} \in R^{C \times H_1 \times W_1}$. C is the number of channels, H and W are the height and width of the feature map, BL is the bilinear interpolation, and T is the transposed convolution.

However, HSFPN has limited capability in perceiving edge information, making it difficult to distinguish between early-stage weed growth and crops. To address this, we propose an Adaptive Hierarchical Feature Fusion Network (AHFPN), which significantly enhances the model's ability to handle multi-scale weed targets and enriches feature representations to improve detection accuracy across different weed growth stages. This module combines the concepts of HSFPN and PAN [31], with improvements tailored to different weed targets. The main process includes:

1) Adding a Conv2d layer to the P4 output to enhance the extraction of high-level semantic information.
2) The output of the P5 layer undergoes CA and a 1x1 Conv2d operation to extract key channel information and adjust weights.
3) The processed high-level features are weighted, multiplied by previously fused features, and then added together.
4) Finally, the fused features pass through the C3k2 module to further enhance feature extraction, resulting in more refined high-level semantic features.

E. LGCD

The detection head in YOLO11 identifies object locations and categories from the feature map. The process is as follows: In the position regression branch, two standard convolutions are used for feature fusion, followed by a convolution layer for location prediction; In the classification branch, depthwise separable convolutions [32] are used for feature fusion, followed by pointwise convolutions for channel-wise information

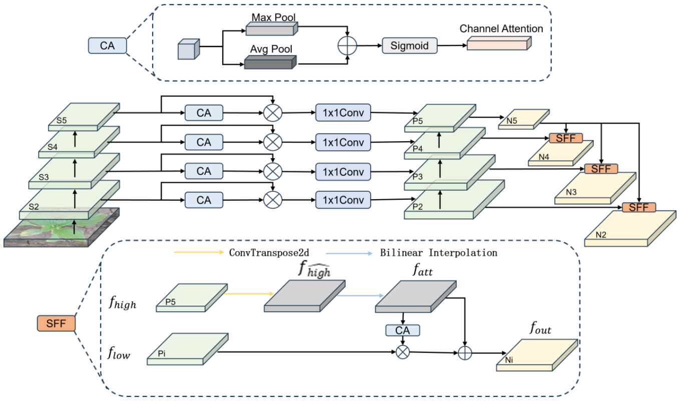

Fig. 4. HSFPN structure and diagram of CA and SFF modules.

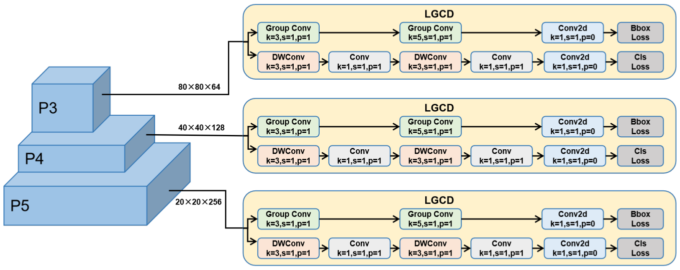

Fig. 5. LGCD structure.

interaction. Finally, a convolution layer performs classification prediction, with a Softmax activation function generating category probabilities. Although YOLO11 is significantly lighter compared to previous YOLO models, it still does not fully meet the real-time and multi-scale requirements for weed detection. Therefore, we propose the lightweight Grouped Convolution Detection Head module, as shown in Fig. 5.

The LGCD module is based on the concept of grouped convolutions, which improves the YOLO11 detection head. In this study, we replace the first two standard convolutions in the position regression branch of the YOLO11n detection head with grouped convolutions to reduce computation and parameter count, achieving the model. To minimize feature loss, we modify the kernel of the second grouped convolution to 5x5, ensuring lightweight while extracting multi-scale weed features and improving detection accuracy.

IV. Experimental Results and Analysis

A. Datasets

CottonWeedDet3 [33] contains 848 RGB images captured in cotton fields in the southern United States, covering three common weed categories: Carpetweed, Morningglory, and Palmer Amaranth. The images capture various angles and natural lighting conditions, ensuring the dataset's diversity and relevance for application. The dataset construction process includes image acquisition, preprocessing, bounding box annotation, data cleaning, format conversion, and data augmentation. Experts manually annotated the images using the SuperAnnotate platform and converted them to the VIA format for further use. To improve data quality, low-quality annotations and out-of-focus areas smaller than 200×200 pixels were cleaned, and erroneous labels were corrected. The final dataset contains 848 images and 1,532 bounding boxes, split into training, validation, and test sets in a 7:1:2 ratio. Fig. 6 shows an example image of the CottonWeedDet3 dataset.

Fig. 6. Example plot of CottonWeedDet3 dataset.

The CottonWeed2 [34] dataset contains 570 images, labeled into two categories: weeds and cotton. The weed category includes various plants, such as Wormwood, Common Sunflower, Chicory, Caltrop, Ginkgo, Castor Bean, Crabgrass, False Sea Purslane, and Amaranthus, reflecting the diversity of weed species. This diversity provides rich and complex samples for model training. Using digital cameras or cellphones, the photos were taken from actual cotton fields in India and stored in.jpg format, which makes them extremely useful. To standardize data processing and adapt to model input, all images were resized to 416 × 416 pixels. This processing method facilitates model handling while reducing computational complexity. The dataset is divided into training, validation, and test sets in a 6:2:2 ratio. Fig. 7 shows an example image of the CottonWeed2 dataset.

Fig. 7. Example plot of CottonWeed2 dataset.

The CottonWeedDet12 dataset [35], collected at Mississippi State University Research Farm, includes 5,648 images and 9,370 bounding box annotations of 12 cotton weed species. Captured under natural light between February and October 2021 using smartphones or handheld cameras, the dataset spans diverse growth stages, lighting, weather, and field conditions to ensure complexity and diversity. The dataset was re-divided using a custom script into training, validation, and test sets at a 7:1:2 ratio. Fig. 8 presents example images from the public CottonWeedDet12 dataset.

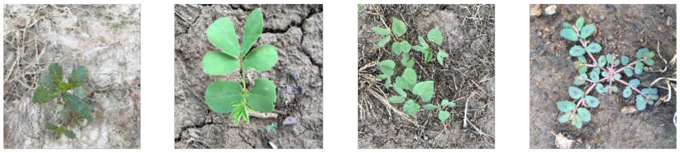

Fig. 8. Example plot of CottonWeedDet12 dataset.

B. Experimental Configuration

The experiment was conducted on a Windows 11 operating system. The model architecture includes Python 3.8.19, PyTorch 2.1.1, and TorchVision 0.16.1, with PyCharm as the integrated development environment. The CPU is an Intel i5-12400F, and the GPU is an Nvidia GeForce RTX 4060Ti (16GB) with 4352 CUDA cores, running CUDA version 12.1.

C. Experimental Parameter Setting and Evaluation Indicators

The model was trained for 300 epochs with a batch size of 8. The AdamW optimizer was used, with an initial learning rate of 0.01 and a momentum of 0.937. The input image size was 640×640. Multiple evaluation metrics were used to assess the effectiveness of this study, including Precision (P), Recall (R), Mean Average Precision (mAP), Parameters (Params), and Giga Floating Point Operations per Second (GFLOPs). The model's recognition performance was measured using IOU thresholds of 0.50 and 0.50:0.95. Params were used to measure the model's parameter count, and GFLOPs were used to measure its computational complexity. The specific definition is given in Eq. (3)-(8):

$$Precision = TP/(TP + FP) \tag{3}$$

$$Recall = TP/(TP + FN) \tag{4}$$

$$AP = \int_0^1 Precision(Recall)dR \tag{5}$$

$$mAP = \sum_{i=1}^{N} APi/N \tag{6}$$

$$GFlops = O\left(\sum_{i=1}^{n} K_i^2 * C_{i-1}^2 * C_i + \sum_{i=1}^{n} m^2 * C_i\right) \tag{7}$$

$$Params = O\left(\sum_{i=1}^{n} M_i^2 * K_i^2 * C_{i-1} * C_i\right) \tag{8}$$

TP represents genuine positives, FP represents false positives, and FN represents false negatives. $Precision$ and $Recall$ refer to the Precision-Recall curve. N represents the number of defects. O represents the constant order, K represents the kernel size, C represents the number of channels, M represents the input image size, and i represents the number of iterations.

D. Ablation Experiments

1) CottonWeedDet3 ablation experiments: To evaluate the performance of the LMS-YOLO11n model in weed detection, ablation experiments were conducted on the CottonWeedDet3 dataset, testing the C3K2_FastGLU, AHFPN, and LGCD modules separately. The results of the ablation experiments are shown in Table I. Compared to the baseline model YOLO11n, LMS-YOLO11n improves mAP50 by 2.5%, while reducing computational load and parameter count by 26% and 37%, respectively.

First, the optimization of C3K2 is discussed. By integrating the FastGLU designed in this study with C3K2, mAP50 increased by 0.9, while both computational load and parameter count were reduced by 6%. This suggests that the C3K2_FastGLU module enhances the model's ability to extract both local and global features of weeds.

Next, the improvement in the NECK section is discussed. After applying the designed AHFPN feature fusion module, mAP50 increased by 0.4%, while computational load decreased by 11% and parameter count by 26%. This demonstrates the effectiveness of the AHFPN module in enhancing weed edge features.

In the detection head, after applying the group convolution-based LGCD module, mAP50 increased by 2.3%, while computational load decreased by 17% and parameter count by 11%. This demonstrates that the LGCD module improves the detailed capture of multi-scale feature information, achieving an optimized balance between feature extraction diversity and model computational efficiency.

Finally, the effect of the cumulative modules was demonstrated. First, combining C3K2_FastGLU with AHFPN resulted in a 0.1% increase in mAP50, with computational load and parameter count decreasing by 0.14% and 32%, respectively. Adding the LGCD module further improved accuracy by 2.5%, while reducing computational load by 26% and parameter count by 37% compared to the baseline model.

2) CottonWeed2 ablation experiments: To validate the model's robustness, ablation experiments were conducted on the CottonWeed2 dataset. The results of the ablation experiments are shown in Table II.

Incorporating C3K2_FastGLU optimized the model's feature extraction capabilities. Compared to the baseline model, mAP50 increased to 75%, while computation (GFLOPs reduced from 6.3 to 5.9) and parameter size (reduced from 2.58 MB to 2.41 MB) decreased. This demonstrates that C3K2_FastGLU enhanced the model's ability to perceive fine-grained weed features and overall contextual information.

The inclusion of AHFPN significantly enhanced feature fusion capabilities, particularly for detecting multi-scale targets. Experimental results showed a mAP50 increase to 75.2%, a 0.7% improvement over the baseline model, with computation and parameter size reduced to 5.6 GFLOPs and 1.89 MB, respectively. This indicates that AHFPN optimized the model structure for selecting and fusing multi-scale features while maintaining computational efficiency.

LGCD focused on improving the fine-grained modeling of multi-scale feature information. Although incorporating this module slightly reduced mAP50 by 0.6%, it decreased computation and parameter size to 5.4 GFLOPs and 2.28 MB, respectively. This highlights its efficiency in reducing redundant computations and enhancing contextual feature fusion, making it well-suited for lightweight and edge device applications.

When C3K2_FastGLU, AHFPN, and LGCD modules were progressively combined, the model exhibited significant synergistic performance improvements. With all three modules combined, the model achieved 79.2% precision, a recall rate of

TABLE I. COTTONWEEDDET3 ABLATION EXPERIMENT TABLE

YOLO11n	C3K2_FastGLU	AHFPN	LGCD	P	R	mAP50	mAP50-95	GFLOPs	Params(MB)
✓				85.8	62.4	73.6	58.4	6.3	2.58
✓	✓			74.7	68	74.5	58.1	5.9	2.41
✓		✓		77.8	67	74	57.8	5.6	1.9
✓			✓	75.1	70.7	75.9	62	5.4	2.28
✓	✓	✓		78.6	68.5	74.6	57.8	5.2	1.73
✓		✓	✓	74	69.4	73.3	58.3	5	1.78
✓	✓	✓	✓	78.2	69.5	76.1	60	4.6	1.61

72%, a mAP50 of 76.4%, and a mAP50-95 of 51.2%, showing comprehensive improvements over the baseline model. Additionally, computation decreased to 4.6 GFLOPs and parameter size to 1.61 MB, demonstrating excellent lightweight performance and resource optimization.

Therefore, by comparing the ablation data from these two datasets, The LMS-YOLO11n model put out in this work may effectively extract the edges and fine-grained characteristics of weeds and meet the needs of deploying in various embedded weed detection devices with real-time requirements.

E. Comparison Experiments

1) CottonWeedDet3 Comparison Experiments with the Latest Models: To comprehensively evaluate the advantages of the proposed LMS-YOLO11n model, several state-of-the-art models, including YOLOv3-tiny, YOLOv5n, YOLOv6n, YOLOv7-tiny, YOLOv8n, YOLOv10n, and YOLO11n, were selected for comparison.The detection performance on the CottonWeedDet3 dataset is compared in Table III, where the bolded text indicates the greatest outcomes. Table III demonstrates that the improved LMS-YOLO11n model outperforms YOLOv3-tiny, YOLOv5n, YOLOv6n, YOLOv7-tiny, YOLOv8n, YOLOv10n, and YOLO11n.The LMS-YOLO11n model achieved a mAP50 of 76.1% and a mAP50-95 of 0.60% on the CottonWeedDet3 dataset, with a computational load of 4.6 GFLOPs and a parameter size of 1.61 MB. While improving accuracy, the model significantly reduced parameter size and computational load. Although the accuracy(P) slightly decreased, mAP50 increased by 2.5%, computational load decreased by 26% and parameter size reduced by 37%.

2) CottonWeed2 Comparison Experiments with the Latest Models: To further verify the generalization capability of LMS-YOLO11n, a comparative experiment was conducted on the CottonWeedDet2 dataset. Table IV presents the comparison results of the latest models on the CottonWeed2 dataset. The LMS-YOLO11n model achieved an mAP50 of 76.4%, and an mAP50-95 of 51.2% on the CottonWeed2 dataset, with a computational load of 4.6 GFLOPs and a parameter size of 1.61 MB.Compared to YOLOv3-tiny, YOLOv5n, YOLOv6n, YOLOv7-tiny, YOLOv8n, YOLOv10n, and YOLO11n, LMS-YOLO11n achieved mAP50 improvements of 8%, 3.1%, 3.0%, 1.3%, 3.1%, 8.4%, and 1.9%, respectively. Additionally, the computational load decreased by 67%, 35%, 60%, 65%, 43%, 28%, and 26%, while the parameter size reduced by 83%, 35%, 61%, 86%, 46%, 21%, and 37%, respectively. These results demonstrate that LMS-YOLO11n achieved the best performance in terms of mAP50, computational load, and parameter size.

3) CottonWeedDet12 Comparison Experiments with the Latest Models: In this paper, comparison experiments are also conducted on the CottonWeedDet12 dataset to further validate the robustness of the LMS-YOLO11n model. The relevant comparison data are shown in Table V. Table V demonstrates the performance comparison of multiple models on the CottonWeed12 dataset, and LMS-YOLO11n stands out in terms of comprehensive performance. YOLOv3-tiny, although having a mAP50 of 91.7%, has a computational and parametric count of 14.3 GFLOPs and 9.52 MB, respectively, which is the model with the highest consumption of computational resources in the table, restricting its application on resource-constrained devices.YOLOv5n and YOLOv6n are optimized in terms of computation volume of 7.1 GFLOPs and 11.5 GFLOPs and number of parameters of 2.5 MB and 4.15 MB, respectively, but their mAP50 values of 92.5% and 90.7% are slightly lower than that of the LMS-YOLO11n. YOLOv7-tiny's mAP50 of 92.7% is still high, but its 13.3 GFLOPs of computation and 12.3 MB of parameter count are similar to that of YOLOv3-tiny. YOLOv8n further optimizes the parameter count and computation with a mAP50 of 92.3%, with values of 3 MB and 8.1 GFLOPs, respectively, but is still not as light as that of LMS- YOLO11n. YOLOv10n and YOLO11n, as more advanced models, exhibit mAP50s of 93% and 93.6%, with their computational and parametric quantities reduced to 6.4 GFLOPs and 6.3 GFLOPs and 2.04 MB and 2.58 MB, respectively. The LMS-YOLO11n, with the minimum computational quantity of 4.6 GFLOPs, and the LMS-YOLO11n achieve the same mAP50 as the YOLO11n with a minimum number of parameters of 1.62 MB. Taken together, the LMS-YOLO11n achieves an optimal balance between performance, efficiency, and lightness with a mAP50 of 93.6%, several parameters of 1.62 MB, and a computation volume of 4.6 GFLOPs, making it suitable for complex field environments and resource-constrained edge device scenarios.

F. Model Detection Effect and Visualization Analysis

The improved LMS-YOLO11n demonstrates excellent detection performance, providing accurate and comprehensive weed recognition under various environmental conditions. HiResCAM [36] was used to perform visualization analysis on the CottonWeedDet3 and CottonWeed2 datasets. In the images, darker colors indicate higher attention, while lighter colors represent lower attention, as shown in Fig. 9 and Fig. 10.

Fig. 9 and Fig. 10 show that both LMS-YOLO11n and YOLO11n can identify and locate target areas dominated by weed structures. However, compared to YOLO11n, LMS-YOLO11n reduces false detections and more accurately focuses on the actual weed shapes. Specifically, YOLO11n

TABLE II. CottonWeed2 Ablation Experiment Table

YOLO11n	C3K2_FastGLU	AHFPN	LGCD	P	R	mAP50	mAP50-95	GFLOPs	Params(MB)
✓				80.9	71.7	74.5	51.8	6.3	2.58
✓	✓			81.4	67.4	75	50.5	5.9	2.41
✓		✓		86.3	66.7	75.2	51.1	5.6	1.89
✓			✓	76.6	67	73.9	51.1	5.4	2.28
✓	✓	✓		84.8	64.3	75	50.1	5.2	1.73
✓		✓	✓	91.4	67	75.1	52.7	5	1.78
✓	✓	✓	✓	79.2	72	76.4	51.2	4.6	1.61

TABLE III. Comparison of Experimental Results of Different Models on CottonWeedDet3 Dataset

	P	R	mAP50	mAP50-95	GFLOPs	Params (MB)
YOLOv3-tiny	78.7	67.4	71.4	52.3	14.3	9.52
YOLOv5n	75.8	64.6	70.6	54.9	7.1	2.5
YOLOv6n	76.6	70.7	74.2	58.6	11.5	4.15
YOLOv7-tiny	85	64.5	73.8	58	13.3	12.3
YOLOv8n	84.2	65.9	74.3	58.4	8.1	3
YOLOv10n	74.7	65.2	70.5	56.1	6.4	2.04
YOLO11n	**85.8**	62.4	73.6	58.4	6.3	2.58
LMS-YOLO11n	78.2	**69.5**	**76.1**	**60.6**	**4.6**	**1.61**

TABLE IV. Comparison of Experimental Results of Different Models on CottonWeed2 Dataset

	P	R	mAP50	mAP50-95	GFLOPs	Params (MB)
YOLOv3-tiny	64.3	71.3	68.1	43.0	14.3	9.52
YOLOv5n	83.9	66.6	72.7	51.1	7.1	2.5
YOLOv6n	**84.5**	65.7	72.6	51.2	11.5	4.15
YOLOv7-tiny	77.6	**72.1**	74.9	45.2	13.3	12.3
YOLOv8n	78.9	70.0	73.3	**53.1**	8.1	3.0
YOLOv10n	83.9	59.5	68.0	43.6	6.4	2.04
YOLO11n	80.9	71.7	74.5	51.8	6.3	2.58
LMS-YOLO11n	79.2	72.0	**76.4**	51.2	**4.6**	**1.61**

may be affected by morphological similarities and lighting variations in complex field environments, leading to scattered feature capture and reduced target localization accuracy. In contrast, LMS-YOLO11n, with its lightweight design and optimized feature extraction modules, effectively suppresses environmental noise and significantly enhances target feature extraction accuracy. It shows a stronger ability to differentiate when weeds and crops have similar morphologies. This demonstrates the superiority and reliability of LMS-YOLO11n for efficient weed detection in complex field scenarios.

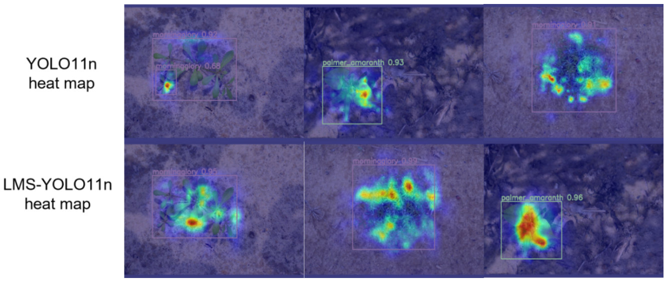

Fig. 9. Contrasting thermal diagrams before and after CottonWeedDet3 model improvement.

TABLE V. Comparison of Experimental Results of Different Models on CottonWeedDet12 Dataset

	P	R	mAP50	mAP50-95	GFLOPs	Params(MB)
YOLOv3-tiny	88.8	86.4	91.7	80.3	14.3	9.52
YOLOv5n	90.6	86.4	92.5	85.3	7.1	2.5
YOLOv6n	90.5	84.8	90.7	84.1	11.5	4.15
YOLOv7-tiny	92.3	86	92.7	82.1	13.3	12.3
YOLOv8n	92.2	85.8	92.3	85.6	8.1	3
YOLOv10n	91.3	87.6	93	**88.2**	6.4	2.04
YOLO11n	**92.4**	86	93.6	87.2	6.3	2.58
LMS-YOLO11n	89.6	**88.9**	**93.6**	86.1	**4.6**	**1.62**

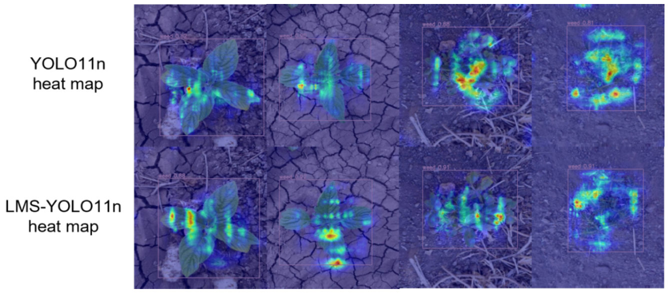

Fig. 10. Contrasting thermal diagrams before and after CottonWeed2 model improvement.

V. Conclusion and Future Work

A. Conclusion

This study introduces LMS-YOLO11n, a novel lightweight weed detection network designed for precision agriculture on edge devices and in complex field environments. The network leverages innovative structural designs to enhance detection accuracy and computational efficiency, making it especially suitable for resource-constrained edge devices.

In the feature extraction phase, LMS-YOLO11n replaces the traditional C3K2 module with the C3K2_FastGLU module, integrating partial convolution and CGLU mechanisms to extract fine-grained weed features more effectively. Compared to traditional convolution methods, FastGLU uses channel-level weighting to enhance sensitivity to fine details, enabling more precise differentiation between weeds and crops in complex field environments. For feature fusion, the study introduces the Adaptive Hierarchical Feature Pyramid Network (AHFPN), which optimizes feature selection and fusion to improve multi-scale weed detection capabilities. AHFPN effectively integrates multi-scale feature maps, enhancing weed detection and preventing the loss of small-scale target information, thereby improving detection accuracy. To boost model efficiency, LMS-YOLO11n replaces the traditional detection head with the lightweight LGCD module. LGCD, designed with group convolutions, reduces parameter and computation requirements while maintaining high detection accuracy, making it ideal for low-power edge devices capable of efficient real-time weed detection.

LMS-YOLO11n demonstrates superior performance across three datasets. On the CottonWeedDet3 dataset, the model achieved a mAP50 of 76.1%. On CottonWeed2, it reached 76.4%, while on CottonWeedDet12, it achieved 93.6%, reducing computation and parameter sizes by 26% and 37%, in contrast to the baseline model, correspondingly. These results demonstrate that LMS-YOLO11n achieves high-precision detection in complex environments and is deployable on edge devices, providing accurate real-time agricultural monitoring solutions.

B. Future Work

Despite the significant results, the proposed model has some limitations:

1) Lack of experiments and deployments in real-world agricultural scenarios. Real agricultural environments involve variations in weed types, densities, and growth states, requiring further validation of the model's performance under these conditions.
2) Detection accuracy for young weeds needs improvement. The simple morphology and texture of young weeds often confuse with the background or crops, leading to limited detection accuracy.

To address these issues, this paper will explore multimodal fusion techniques in future research to solve the challenges of young weed detection. By fusing different types of data sources, it can provide richer feature information for the model and help it recognize young weeds more accurately. Meanwhile, more field experiments are planned to be conducted in combination with practical agricultural application scenarios to test the performance of the model in different environments, to further improve its adaptability and accuracy, and to provide more effective solutions for precision agriculture.

Acknowledgment

This work was supported by the Programs for Natural Science Foundation of Xinjiang Uygur Autonomous Region, Grant number 2022D01C54.

References

[1] Bah, M. D., Hafiane, A., Canals, R., & Emile, B. (2019, November). Deep features and One-class classification with unsupervised data for weed detection in UAV images. In *2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA)* (pp. 1-5). IEEE.

[2] Donayre, D. K. M., Santiago, S. E., Martin, E. C., Lee, J. T., Corales, R. G., Janiya, J. D., & Kumar, V. (2019). Weeds of Vegetables and other Cash Crops in the Philippines. *Philippine Rice Research Institute, Maligaya, Science City of Muñoz, Nueva Ecija. 141pp.*

[3] Zhang, W., Miao, Z., Li, N., He, C., & Sun, T. (2022). Review of current robotic approaches for precision weed management. *Current robotics reports*, *3*(3), 139-151.

[4] Jin, X., Liu, T., Chen, Y., & Yu, J. (2022). Deep learning-based weed detection in turf: a review. *Agronomy*, *12*(12), 3051.

[5] Wang, A., Peng, T., Cao, H., Xu, Y., Wei, X., & Cui, B. (2022). TIA-YOLOv5: An improved YOLOv5 network for real-time detection of crop and weed in the field. Frontiers in Plant Science, 13, 1091655.

[6] Kumar, D. A., & Prema, P. (2016). A NOVEL WRAPPING CURVELET TRANSFORMATION BASED ANGULAR TEXTURE PATTERN (WCTATP) EXTRACTION METHOD FOR WEED IDENTIFICATION. ICTACT Journal on Image & Video Processing, 6(3).

[7] Sujaritha, M., Annadurai, S., Satheeshkumar, J., Sharan, S. K., & Mahesh, L. (2017). Weed detecting robot in sugarcane fields using fuzzy real time classifier. Computers and electronics in agriculture, 134, 160-171.

[8] Chen, X., & Gupta, A. (2017). An implementation of faster rcnn with study for region sampling. arxiv preprint arxiv:1702.02138.

[9] Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence, 39(6), 1137-1149.

[10] Zhang, X., Cui, J., Liu, H., Han, Y., Ai, H., Dong, C., ... & Chu, Y. (2023). Weed identification in soybean seedling stage based on optimized Faster R-CNN algorithm. Agriculture, 13(1), 175.

[11] Özcan, R., Tütüncü, K., & Karaca, M. (2022). Comparison of Plant Detection Performance of CNN-based Single-stage and Two-stage Models for Precision Agriculture. Selcuk Journal of Agriculture and Food Sciences, 36(4), 53-58.

[12] Li, Z., Li, Y., Yang, Y., Guo, R., Yang, J., Yue, J., & Wang, Y. (2021). A high-precision detection method of hydroponic lettuce seedlings status based on improved Faster RCNN. Computers and electronics in agriculture, 182, 106054.

[13] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14 (pp. 21-37). Springer International Publishing.

[14] Redmon, J. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition.

[15] Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., ... & Wei, X. (2022). YOLOv6: A single-stage object detection framework for industrial applications. arxiv preprint arxiv:2209.02976.

[16] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 7464-7475).

[17] Wang, C. Y., Yeh, I. H., & Mark Liao, H. Y. (2025). Yolov9: Learning what you want to learn using programmable gradient information. In European Conference on Computer Vision (pp. 1-21). Springer, Cham.

[18] Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J., & Ding, G. (2024). Yolov10: Real-time end-to-end object detection. arxiv preprint arxiv:2405.14458.

[19] Chen, J., Wang, H., Zhang, H., Luo, T., Wei, D., Long, T., & Wang, Z. (2022). Weed detection in sesame fields using a YOLO model with an enhanced attention mechanism and feature fusion. Computers and Electronics in Agriculture, 202, 107412.

[20] Zhao, Y., Sun, C., Xu, X., & Chen, J. (2022). RIC-Net: A plant disease classification model based on the fusion of Inception and residual structure and embedded attention mechanism. computers and Electronics in Agriculture, 193, 106644.

[21] Hong, W., Ma, Z., Ye, B., Yu, G., Tang, T., & Zheng, M. (2023). Detection of green asparagus in complex environments based on the improved YOLOv5 algorithm. Sensors, 23(3), 1562.

[22] Liu, L., Liang, J., Wang, J., Hu, P., Wan, L., & Zheng, Q. (2023). An improved YOLOv5-based approach to soybean phenotype information perception. Computers and Electrical Engineering, 106, 108582.

[23] Zhang, C., Liu, J., Li, H., Chen, H., Xu, Z., & Ou, Z. (2023). Weed Detection Method Based on Lightweight and Contextual Information Fusion. Applied Sciences, 13(24), 13074.

[24] Guo, A., Jia, Z., Wang, J., Zhou, G., Ge, B., & Chen, W. (2024). A lightweight weed detection model with global contextual joint features. Engineering Applications of Artificial Intelligence, 136, 108903.

[25] Fan, X., Sun, T., Chai, X., & Zhou, J. (2024). YOLO-WDNet: A lightweight and accurate model for weeds detection in cotton field. Computers and Electronics in Agriculture, 225, 109317.

[26] Chen, Y., Zhang, C., Chen, B., Huang, Y., Sun, Y., Wang, C., ... & Gao, Y. (2024). Accurate leukocyte detection based on deformable-DETR and multi-level feature fusion for aiding diagnosis of blood diseases. Computers in Biology and Medicine, 170, 107917.

[27] Chen, J., Kao, S. H., He, H., Zhuo, W., Wen, S., Lee, C. H., & Chan, S. H. G. (2023). Run, don't walk: chasing higher FLOPS for faster neural networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 12021-12031).

[28] Shazeer, N. (2020). Glu variants improve transformer. arxiv preprint arxiv:2002.05202.

[29] Shi, D. TransNeXt: Robust Foveal Visual Perception for Vision Transformers. arxiv 2023. arxiv preprint arxiv:2311.17132.

[30] Wang, W., **e, E., Song, X., Zang, Y., Wang, W., Lu, T., ... & Shen, C. (2019). Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 8440-8449).

[31] Howard, A. G. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arxiv preprint arxiv:1704.04861.

[32] Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1492-1500).

[33] Rahman, A., Lu, Y., & Wang, H. (2022). Deep neural networks for weed detections towards precision weeding. In 2022 ASABE Annual International Meeting (p.1). American Society of Agricultural and Biological Engineers.

[34] Kumaran, D.T. Cotton-Weed Dataset. (2021). Available online: https://universe.roboflow.com/deepak-kumaran-t/cotton-weed.

[35] Dang, F., Chen, D., Lu, Y., & Li, Z. (2023). YOLOWeeds: A novel benchmark of YOLO object detectors for multi-class weed detection in cotton production systems. Computers and Electronics in Agriculture, 205, 107655.

[36] Draelos, R. L., & Carin, L. (2020). Use HiResCAM instead of Grad-CAM for faithful explanations of convolutional neural networks. arxiv preprint arxiv:2011.08891.