UDC 004.93

LONG-DISTANCE CABBAGE DAMAGE AND PEST DETECTION METHOD USING YOLO11

Khabarlak K. S. – PhD, Associate Professor of the Department of System Analysis and Control, Dnipro University of Technology, Dnipro, Ukraine. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0003-4263-0871.

Laktionov I. S. – Dr. Sc., Full Professor, Professor of the Department of Computer Systems Software, Dnipro University of Technology, Dnipro, Ukraine. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0001-7857-6382.

Gorev V. N. – PhD, Associate Professor, Head of the Department of Physics, Dnipro University of Technology, Dnipro, Ukraine. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0002-9528-9497.

Diachenko G. G. – PhD, Associate Professor of the Department of Electric Drive, Dnipro University of Technology, Dnipro, Ukraine. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0001-9105-1951.

ABSTRACT

Context. To ensure sustainable yield, plant health must be constantly monitored with timely measures applied to prevent disease spread. Traditional approaches rely on manual inspection of plants, while neural networks require large amounts of annotated data to train. Both manual inspection and data annotation require expert knowledge and are time-consuming. Close-up photos of leaves are often used for training as they are easier to collect from the Internet. However, this complicates disease spread estimation at a scale. Cabbage is one of the plants widely grown in Ukraine, but existing research focusing on cabbage health monitoring is limited.

Objective. The goal of this work is to build a neural-network-based cabbage disease and pest detection system, which can be trained in on a small number of training images. At inference the system should detect pests on plant images at a distance of a whole plant.

Method. Given that existing plant disease datasets, such as IP102 and PlantDoc mostly contain close-up images of diseased plants, the networks trained on such datasets suffer from lack of generalization to images at a distance. To select the best object detection model, state-of-the-art object detection architectures, namely YOLO 8, 9, 10, 11, and RT-DETR have been analyzed in the work. To increase detection distance multi-image loss is proposed to improve hyperparameter search using Tree-Structured Parzen Estimators. Also, to improve detection quality, a novel cabbage disease dataset has been collected in Dnipro region, Ukraine. The new classes include crucifier flea beetle (widespread pest in Dnipro region) and damaged leaf. When the pest is not visible, but leaf damage is taken, determining specific pest might not be possible. Therefore, we introduce additional damaged leaf class, that captures generic plant damage. This also enables tracking of plant healing rate, when measures to stop pest spread have been taken. We combine collected images with the larger IP102 dataset to increase the number of pests covered to form new Cabbage+IP102 dataset.

Results. 1) Tree-Structured Parzen Estimators search on the multi-image loss has improved the YOLO 11 M performance from 0.3642 to 0.3892 mAP50-95 on images taken at a distance. 2) Collected dataset has enabled detection of cabbage plant health problems at a distance, including cases when the pest is currently not visible, but the damage is present.

Conclusions. In this work, the cabbage pest and damaged leaf YOLO 11 M detection system has been presented. The detector architecture has been selected as the best-found during analysis on 2 datasets. The developed system requires only 7 annotated cabbage images to be trained and to perform pest and damaged leaf detection on high resolution images (2016x2016) of whole cabbage plants. The final model can be used to monitor cabbage health problems, damage, and rate of healing using images taken at a distance.

KEYWORDS: agriculture, deep learning, plant health monitoring, pest detection, leaf damage estimation, yolo 11, brassicaceae family.

ABBREVIATIONS

YOLO is a You Only Look Once family of single-stage object detection models;

RT-DETR is a Real-Time Detection Transformer, a transformer-based object detection model;

TPE is a Tree-Structured Parzen Estimators hyperparameter optimization method.

NOMENCLATURE

M is an input image of shape $W \times H \times C$, where W,H,C are width, height and the number of color channels;

$\Phi(x)$ is a detection neural network, that takes image M as an input and returns $\{D\}_{i=0}^{\mathrm{Max}_{\mathrm{det}}}$ predictions;

$\{D\}_{i=0}^{\mathrm{Max}_{\mathrm{det}}}$ is a maximum number of possible detections for a given network;

D_i is a tuple with object detection information, which typically takes form ($Shape_i$, $ObjectnessScore_i$, $ClassProbabilities_i$);

f is a training loss function;

$\mathrm{mAP}_{50\text{–}95}$ is a mean average precision detection metric;

$p(x|y)$ is a TPE hyperparameter kernel density estimate.

INTRODUCTION

Increasing plant yields per hectare is an important agricultural problem. Traditional methods of plant health monitoring rely on manual inspections and visual diag-

DOI 10.15588/1607-3274-2026-1-4

nostics, which can be time-consuming and labor-intensive, particularly over large farmlands. In recent years hardware and software assisted technologies are widely used to monitor plant growing conditions and to take timely actions [1]. The approaches include land fertilization, spraying and irrigation of fields using autonomous vehicles or drones, climate and soil parameters monitoring, while recent advancements in computer vision and machine learning have provided promising solutions for automating the detection of plant pests and diseases.

Machine learning-based plant health problem detection is typically performed on datasets that capture short-distance images of pests [2–5]. There are several reasons for that: 1) such datasets can be collected from images found on the Internet; 2) both annotating images in short-distance datasets and training a neural network to detect large disease manifestation are simpler to perform. However, in the field conditions it would have been beneficial to automatically detect the presence of illness at a distance, which requires the use of appropriate datasets and improved training techniques.

Annotating images of a large field with instances of disease manifestations (e.g., malicious insects) spanning only several pixels is a complicated problem, that requires significant time from experts to annotate the dataset. Cabbage is one of the plants widely grown in Ukraine, but existing research focusing on cabbage health monitoring is limited.

Therefore, in this work we investigate the use of state-of-the-art general-purpose detection neural networks for the task of long-distance cabbage plant health monitoring and propose a method of model hyper-parameter selection based on Tree-Structured Parzen Estimators.

The object of this study is the process of automatic image-based cabbage plant health monitoring.

The purpose of this study is object detection neural networks and hyperparameter selection algorithms.

The purpose of the research is to build a neural-network-based cabbage disease and pest detection system, which is capable of accurate pest detection based on image of the whole plant.

1 PROBLEM STATEMENT

Let M be an input image tensor of shape $W \times H \times C$, where W, H, C are width, height and number of channels. For RGB image C=3; let $\Phi(x)$ be a detection neural network, that takes image M as an input, and returns $\{D\}_{i=0}^{\mathrm{Max}_{\mathrm{det}}}$ tuples with information about predicted object bounding boxes, which typically takes form ($Shape_i, ObjectnessScore_i, ClassProbabilities_i$), where *Shape* is defined depending on detection model as *(left,top,width,height)* or ($center_x, center_y, width, height$) object bounding box coordinates, *ObjectnessScore* is a score whether this bounding box contains actual object, *ClassProbabilities* is a vector R^K of class probabilities, where K is the number of classes.

The goal is to minimize loss function f, that defines distance between true and predicted *Shape*, *ObjectnessScore*, and *ClassProbabilities*. The exact loss function f is different between detection neural network models.

2 REVIEW OF THE LITERATURE

To ensure sustainable yield, plants must be constantly monitored for diseases with timely measures applied to prevent disease spread. Traditional approaches rely on manual inspection of plant leaves and trunks to detect any signs of a virus, fungi, bacteria or pests. Obviously, this approach is time-consuming, not scalable to large farmlands, moreover early development of illness is easy to miss. Therefore, the focus of recent agricultural research is shifted towards automation of plant health monitoring.

Automated disease detection approaches are based either on monitoring weather conditions and predicting probability of appearance of certain disease [1, 6] or by capturing images of plants via a smartphone or drone [1, 7]. Both approaches have certain benefits and disadvantages. While weather monitoring is suitable for predicting illness before it occurs, it works with typicality and cannot detect spontaneous illness spreads. In contrast, computer-vision-based methods detect existing plant health problems and can distinguish between different kinds of illnesses, so the most effective treatment could be selected for the specific disease or pest. In this work we focus on computer-vision-based methods such as those that are more precise in disease or pest detection.

Typically, image-based plant health problem detection is formulated as classification or detection. In classification formulation, given a close-up image of plant leaf, trunk or fruit the task is to say what specific kind of illness there is or that no illness is present [8]. Obviously, in this case finding regions of interest remains manual labor. To automatically find the disease, detection methods should be used.

Neural networks have shown high quality in solving detection of arbitrary objects. Early approaches were 2-stage: region proposal and subsequent classification of the proposed regions, like R-CNN [9], Faster R-CNN [10]. Recent research has mostly shifted towards single-stage approaches, like YOLO family of models [11–13] or DETR [14]. Single stage approaches generally have faster inference and higher quality.

Many research approaches were proposed in the field of plant disease and pest detection. The authors of [2] propose a modified YOLOv7-tiny architecture for plant pest detection. The authors target environment with constrained computational resources, therefore, select the smallest variation of the YOLOv7 family of models to improve upon. The resulting model is 47% faster, while offering a 3% improvement in mAP50 over the base YOLOv7-tiny model. Training and testing are performed on images of 21 pests, which is a subset selected from the IP102 [15] dataset. In [5] YOLOv5M has been improved for pest detection by introducing SwinTransformer-based blocks.

DOI 10.15588/1607-3274-2026-1-4

In [16] a modification of YOLOv10 is introduced. The authors reduce computation complexity of the network, while improving tomato fruit detection and ripeness estimation. The authors focus on detection of overlapped and partially occluded fruits.

In [4] the authors note an importance of cabbage (Brassicaceae family) pests and diseases detection for sustained yield and indicate that the field is largely unexplored. In the paper the authors: 1) collect a novel dataset, that consists of 21 classes (14 pests, 3 damage symptoms, 4 beneficial insects); 2) evaluated YOLOv5, 7–11 models on the proposed dataset.

In [3] the authors note the difficulty of detecting large number of small pests. To resolve the issue, they 1) collect a dataset of pests using a light trap; 2) propose an improved R-CNN architecture with novel feature fusion block to improve small pest detection.

Overall neural networks are known to require large amounts of annotated data to be trained. While annotating images with commodity objects can be outsourced to a crowdsourcing platform, annotation of plant disease requires expert knowledge. Thus, acquiring additional training images requires significant effort from human experts, and therefore hard and expensive to acquire. Few-shot learning approaches attempt to resolve the problem. Main few-shot learning approaches include [17]: metric learning, where embedding into a metric space is learned, that is general enough to quickly accommodate to new classes; optimization-based, where a special optimization procedure is defined, that trains "generic" model weights, that are easy to fine-tune; model-based, where model architecture is changed to include memory cells. Transfer learning can also be used in a few-shot scenario, where learning is performed in 2 stages: training on a large general-purpose dataset, then finetuning on a smaller target dataset. Transfer learning is widely used due to its simplicity and sufficiently good results. Hence, in this work we focus on using transfer learning for few-shot neural network training.

Several few-shot learning approaches for plant pest recognition have been proposed. In [18] metric-based few-shot learning plant disease classification method is introduced. Image embeddings are generated first by using ResNet-18 convolutional neural network and then refined by a Transformer. Contrastive loss is used to perform the training. Mahalanobis distance is calculated between new image and existing classes to perform the classification.

The authors of [19] focus on few-shot learning of plant pests. Pre-trained ResNet-50 backbone is used as a part of Faster R-CNN 2-stage detection neural network. To improve working with images of different scale multi-input single-output feature pyramid network is used.

However, the distance at which plant health problem detection is typically performed (typically leaf or part of a plant) is still quite small, which complicates disease spread estimation at a scale. Also, newer detection architectures like YOLO 8, 9, 10, 11 presented in [20–23] or RT-DETR [24] (a modification of DETR) still remain mostly unexplored in the field of plant illness and pest detection on open dataset. Research of cabbage (Brassicaceae family) plant health issue detection is underrepresented in literature.

Therefore, in-the-field plant health monitoring is an open research problem. In this work we: 1) propose a method for the model hyper-parameter tuning for efficient training with few training samples; 2) collect a few-shot dataset for long-distance monitoring of cabbage plant pests and leaf damage.

3 MATERIALS AND METHODS

A generalized scheme of steps taken to improve large distance cabbage health problem detection is shown in Fig. 1.

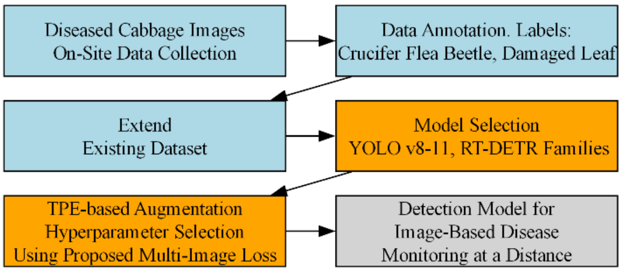


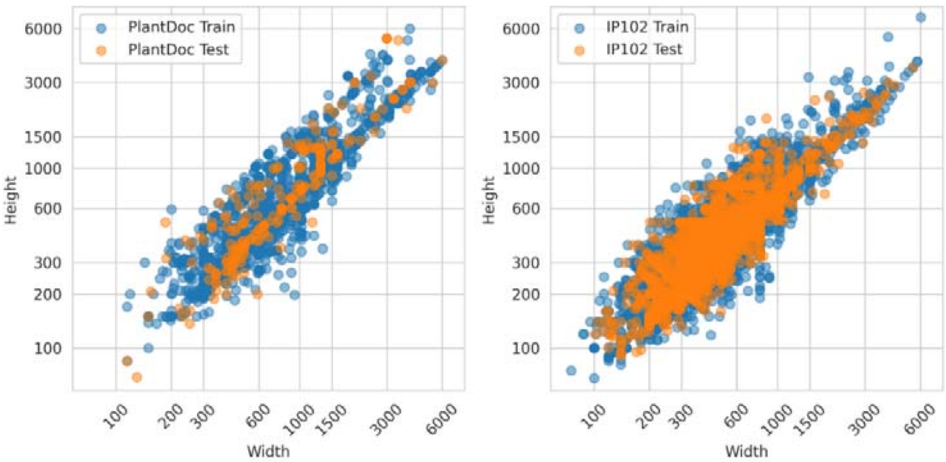
Figure 1 – Generalized scheme of research

First, we analyze existing plant pest and disease open datasets. Then collect in-the-field images of cabbage and annotate them with 2 new classes: crucifer flea beetle (which is widespread in Dnipro region in 2024/25) and damaged leaf. The latter is useful when the pest is not visible, but leaf damage is taken. Also, plant healing can be tracked after measures to stop pest spread have been taken. We combine collected images with the larger IP102 dataset to increase the number of pests that can be detected.

Next, we present a comprehensive analysis of YOLO 8–11 and RT-DETR architectures on 2 widely used plant health detection datasets, namely PlantDoc [25] and IP102 [15]. We perform transfer learning of these models. Each model is considered in different sizes available (nano, medium, large, etc., depending on the model), and provide guidance on selecting the best model.

Finally, we propose multi-image validation loss for searching model hyperparameters using Tree-Structured Parzen Estimators to improve the quality of model training on images taken at a distance. Joined Cabbage+IP102 dataset is used. The final trained model can not only perform detection not only on close-up photos of leaves, but also on plant pictures at a distance (e.g., full cabbage plant).

DOI 10.15588/1607-3274-2026-1-4



(a) PlantDoc (b) IP102

Figure 2 – Distribution of image resolutions present in (a) PlantDoc and (b) IP102 datasets (Log-Log scale)

Many datasets contain images of detached diseased leaf on white background, which does not correspond to real world conditions. In this paper we consider PlantDoc and IP102 datasets, that contain in-the-wild images. PlantDoc [25] has 2 flavors: detection (with 1 or multiple objects per image with bounding boxes) or classification (images cropped to the bounding box). It has 17 classes, and on average, 3.5 bounding boxes per image. IP102 [15] dataset, which has 102 classes of pests and 7.3 times more annotated images. It also has classification and detection sets. On average, it has 1.2 bounding boxes per image, which is less than PlantDoc 3.4 objects per image.

Distribution of image resolutions of the two datasets is shown in Fig. 2 (Log-Log scale). Both datasets contain vertical, horizontal and square images. Although at resolutions above 1500 pixels in either dimension square images are less present. Most images have resolutions below 1500x1500 pixels. Low (below 400 pixels) and very low (below 200 pixels) resolution images are present in both datasets.

While IP102 and PlantDoc datasets offer a strong baseline for training plant health-related problems, they have some disadvantages:

1. PlantDoc contains images either of healthy leaves or damaged by infectious diseases (bacteria, virus, fungi). Images of pests are not present.

2. IP102 dataset has been constructed by collecting images from the Internet and then annotating them. So mostly, the dataset contains close-up images of pests.

For plant pest detection in real-world environment on a farm in Dnipro region, Ukraine, we have found the following challenges in using these datasets for training:

1. It is desirable to capture large images of plants (e.g., whole cabbage plant) and estimate the number of pests on it. In this case the pests appear to be quite small and numerous, which is different from these datasets.

2. Images of not every pest can be captured during worktime. Some are active, for instance, very early in the morning (e.g. slugs) and only damaged leaves can be observed after the fact.

To resolve these problems, we have collected a small dataset with images of cabbage with health problems for few-shot training. Overall, 10 images, 7 in the train set, 3 in the test set. These images have been annotated with 2 classes: crucifer flea beetle (Phyllotreta Cruciferae) and damaged leaf. The dataset has a high number of object instances: 345 instances of crucifer flea beetle and 134 instances of damaged leaf. Collected dataset information is shown in Table 1. It was farmers' demand to perform detection of this beetle as in farm in Dnipro region, Ukraine in 2024 the crucifer flea beetle was widespread, and early detection was important to save the yield of cabbage. Annotated samples are shown in Fig. 3, crucifier flea beetle is highlighted with violet bounding boxes, and leaf damage with yellow.

Table 1 – Collected dataset information.

Class	# train images	# train object instances	# test images	# test object instances
Crucifer Flea Beetle	5	201	3	144
Damaged Leaf	7	91	3	43

To give the neural network knowledge about different kinds of pests given a small training dataset of cabbage images, we extend the IP102 dataset, which is the largest pest dataset to the best of our knowledge. Summary of datasets is presented in Table 2. Cabbage+IP102 is the extended version of the dataset with our annotated images.

DOI 10.15588/1607-3274-2026-1-4

Figure 3 – Examples of the collected and annotated images

Table 2 – Comparison of detection datasets used for training

Dataset	# train	# test	# classes	# objects per image
IP102	15178	3798	102	min: 1, mean: 1.2, max: 26
Cabbage+ IP102	15185	3801	104	min: 1, mean: 1.2, max: 86
PlantDoc	2348	237	17	min: 0, mean: 3.4, max: 42

It should be noted that IP102 dataset contains flea beetle class. However, pests on collected images of cabbage were not detected as such. During deeper investigation, we have found out the IP102 dataset contains photos of other species of flea beetle (Phyllotreta vittula) and macro photos were taken.

To evaluate model performance, we use mAP$_{50–95}$ (mean average precision), which is a common choice. The following formula can be used to compute the metric

$$\mathrm{mAP}_{50-95} = \frac{1}{K \cdot N_{\mathrm{IoU}}} \sum_{k=1}^{K} \sum_{i=1}^{N_{\mathrm{IoU}}} \mathrm{AP}_{k,\mathrm{IoU}_i}\ , \quad (1)$$

where K is the number of classes, N_{IoU} is the number of intersections over union thresholds (IoU), IoU is taken from 0.5 to 0.95 at 0.05 steps, $\mathrm{AP}_{k,\mathrm{IoU}_i}$ is the average precision for class k at the IoU threshold IoU_i.

Both PlantDoc and IP102 datasets contain close-up images (in certain cases, macro images) of pests, and typically only one or few pests are visible. It has been shown that searching augmentations is an effective [26] way to improve the trained model quality. In [12] it has been shown that by applying mosaic augmentation during training, model detection performance can be significantly improved.

In this work we propose to construct hyperparameter optimization loss using multiple images to improve model performance on images captured at a distance:

1. Validation images are rescaled to the resolution of $W_{single} \times H_{single}$.

2. 4×4 grid of resolution $W_{grid} \times H_{grid}$ is formed from these images.

3. Given landmark locations for image at row i and column j:

$$(centerX_{(i,j)}, centerY_{(i,j)}, width_{(i,j)}, height_{(i,j)}).$$

Landmark locations are updated as follows:

$$\begin{aligned} centerX'_{i,j} &= \frac{centerX_{i,j} \cdot W_{\mathrm{single}}}{W_{grid}} + offsetX_{i,j}, \\ centerY'_{i,j} &= \frac{centerY_{i,j} \cdot H_{\mathrm{single}}}{H_{grid}} + offsetY_{i,j}, \\ width'_{i,j} &= \frac{width_{i,j} \cdot W_{\mathrm{single}}}{W_{grid}}, \\ height'_{i,j} &= \frac{height_{i,j} \cdot H_{\mathrm{single}}}{H_{grid}}, \end{aligned} \quad (2)$$

where $offsetX_{i,j}$, $offsetY_{i,j}$ are defined as follows:

$$\begin{aligned} offsetX_{i,j} &= j \cdot W_{\mathrm{single}}, \\ offsetY_{i,j} &= i \cdot H_{\mathrm{single}}. \end{aligned} \quad (3)$$

4. mAP$_{50–95}$ using formula (1) is computed on constructed images.

In this work $W_{single}=H_{single}=640$, $W_{grid}=H_{grid}=2560$.

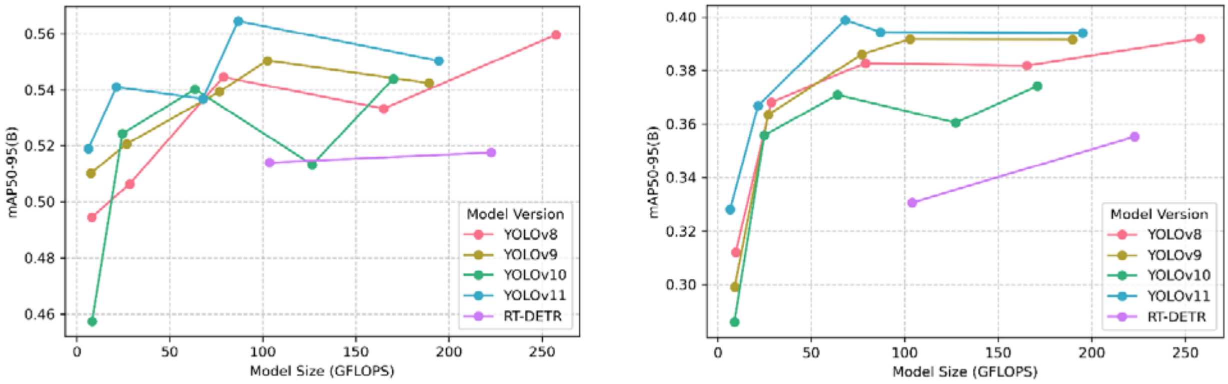
To perform hyperparameter search we use Tree-Structured Parzen Estimators (TPE) [27]. The approach has been to be effective for hyperparameter optimization and is widely used [28].

TPE models search space for the best hyperparameters using kernel density estimation of “bad” hyperparameters $l(x)$ and “good” hyperparameters $g(x)$. Given a minimization problem, the general density $p(x|y)$ is defined as follows [27]:

$$p(x \mid y) = \begin{cases} l(x), & \text{if } y < y^*. \\ g(x), & \text{if } y \geq y^*. \end{cases} \quad (4)$$

where $\{x_i\}$ are hyperparameter observations, $\{y_i\}$are the corresponding loss function values, y^* is a quantile γ of the best observed y values, such that $p(y<y^*)= \gamma$.

DOI 10.15588/1607-3274-2026-1-4



(a) PlantDoc (b) IP102

Figure 4 – Model mAP50–95 depending on model size at 25 training epochs

4 EXPERIMENTS

First, experiments to select the best model for plant pest and disease detection are conducted. We consider YOLO 8, 9, 10, 11 and RT-DETR families of models in different sizes as is shown in Table 3. Size abbreviations are the following: N – nano, T – tiny, S – small, M – medium, C – compact, L – large, X – extra-large, E – extended.

To train the models we use image resolution of 640x640 pixels, which is typically used for pretraining the detection models, and is reasonable for image resolutions of PlantDoc and IP102 datasets (based on Fig. 2). Results are shown on the test set. The batch size is 64, the number of training epochs is set to 25. To improve the quality of detection models, augmentations are used, which include color augmentations (random variations of color hue, saturation, value). Mosaic augmentation [12] is applied during training to improve model generalization. It combines several images into one and has been shown to improve detection quality. Finally, horizontal flip with probability 0.5 is used.

Table 3 – Model sizes considered during the analysis

Model Family	Sizes Considered
YOLO v8	N, S, M, L, X
YOLO v9	T, S, M, C, E
YOLO v10	N, S, M, L, X
YOLO 11	N, S, M, L, X
RT-DETR	L, X

Next, training speed experiments are conducted to see which models require less training steps to achieve high quality. Training is performed for 1, 2, 5, 10, 15, 20, 25, 50 epochs. These experiments are conducted on open IP102 and PlantDoc dataset, so that the analysis can be reused by other researchers.

Finally, augmentation hyperparameter search is used to improve model performance on Cabbage+IP102 dataset. Multi-image validation loss is used for TPE optimization. This dataset has extra images with a high number of annotations per image. We use the best model selected in previous analysis and improve it by performing augmentation hyperparameter search. 60 iterations of hyperparameter search are performed. Parameter ranges considered are defined in Table 4. The probability of horizontal flip augmentation is fixed at 0.5, which is standard value, and is not optimized.

Table 4 – Hyperparameter search ranges

Hyperparameter	Values
HSV: Hue	[0.0, 0.1]
HSV: Saturation	[0.0, 0.9]
HSV: Value	[0.0, 0.9]
Rectangular Images	[False, True]
Multiscale Training	[False, True]
Disable Mosaic Last Epochs	[0, 25]
Degrees Rotation	[0.0, 90.0]
Probability of Flip Upside Down	[0.0, 0.5]
Mosaic Probability	[0.0, 1.0]

5 RESULTS

In Fig. 4 we show $mAP_{50–95}$ of all considered models on PlantDoc and IP102 datasets. On x axis are glops, and not model size names, because the number of floating-point operations is different across model versions. On PlantDoc dataset (Fig. 4a) the test mAP50–95 results are more noisy, likely due to smaller number of test images. Still, YOLO 11 shows one of the best results. On IP102 dataset (Fig. 4b), YOLO 11 models show the best results in each size, while RT-DETR shows the worst. On this dataset YOLO 11 M has the best result on the test set. RT-DETR model underperforms on both datasets, showing the worst mAP to gflops ratio

Note that the smallest variant (nano for YOLO 8,10,11 or tiny for the 9) has on average 67% reduction of gflops and 16% reduction of mAP50–95 with respect to small size. This is a higher quality degradation rate than between other adjacent model sizes. For instance, small variants have 65% gflops reduction over medium, but only 5% quality loss.

In Figs. 5 and 6 we show box plots plot for each of the model versions depending on the number of training epochs for the PlantDoc and IP102 datasets correspondingly. A single box plot is built based on training results of different-sized models of the specified model family.

DOI 10.15588/1607-3274-2026-1-4

As can be seen, in most cases there is mAP outlier below each box plot. The outlier corresponds to the smallest variation (nano or tiny) of this model version.

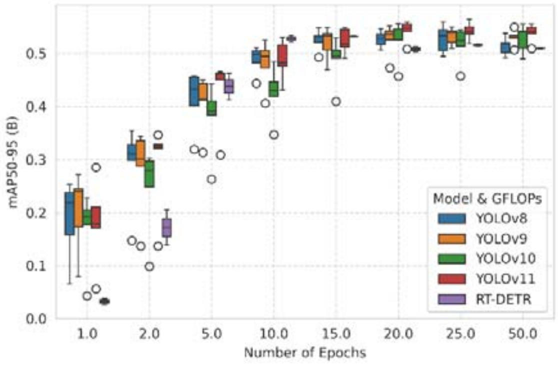


Figure 5 – PlantDoc dataset: model $mAP_{50–95}$ depending on the number of epochs

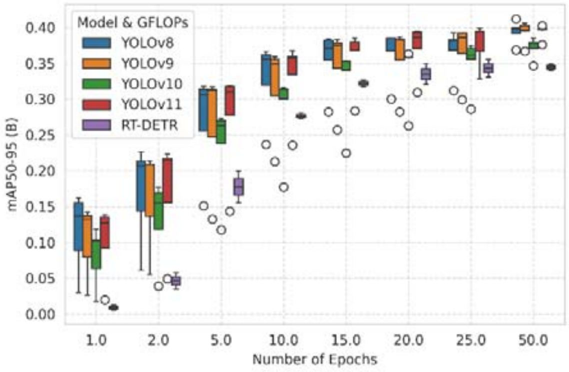


Figure 6 – IP102 dataset: model $mAP_{50–95}$ depending on the number of epochs

On both datasets training for more than 15–20 epochs doesn't result in substantially improved mAP for all models except the smallest ones. When training for a small number of epochs, the best results are obtained from the larger models.

Augmentation hyperparameter optimization values are shown in Table 5, and final training and validation accuracy in Table 6. Hyperparameter optimization procedure has been performed for 60 iterations. The best value has been achieved on iteration 19.

6 DISCUSSION

Best-found validation hyperparameters, found using the proposed multi-image validation loss, have several important changes, when compared to the default augmentation values proposed by the Ultralytics library. Both color saturation and value augmentations were reduced during the search process. During further inspection, high changes in saturation or value result in pests being much harder to detect by human observers. This is explained by the fact that many pests mimic the color of plants on which they live. Also, pest color is important to distinguish between similar species of pests.

Table 5 – Augmentation hyperparameters of the initial and best-found models

Augmentation Hyperparameter	Initial	Best
HSV: Hue	0.0150	0.0240
HSV: Saturation	0.7000	0.3367
HSV: Value	0.4000	0.0108
Rectangular Images	False	False
Multiscale Training	False	False
Disable Mosaic Last Epochs	10	8
Degrees Rotation	0.00	16.03
Probability of Flip Upside Down	0.0000	0.0760
Probability of Horizontal Flip (not searched)	0.5	0.5
Mosaic Probability	1.0000	0.5392

Table 6 – mAP50–95 comparison of the initial and best-found models

Dataset	mAP50–95 (Train)	mAP50–95 (Validation)
Cabbage+ IP102	0.3116	0.3642
Cabbage+ IP102+multi-image loss	0.2901	**0.3892**

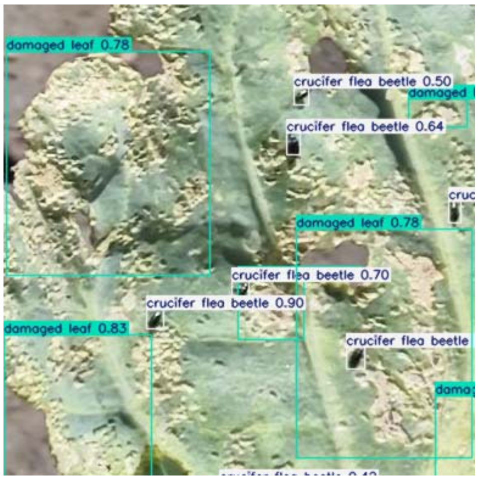
Interestingly, using rectangular images or multiscale training doesn't improve quality on the multi-image validation dataset. By default (Rectangular Image = False), all training images are squeezed into square. If rectangular images parameter is enabled, during training images are resized to the most common aspect ratio. Validation images always use rectangular shapes. Multiscale training uses images of different sizes during training; actual input image resolution is different from batch to batch. Not to be mistaken with resize augmentation, where input image size is the same, but the image is randomly scaled. However, in practice, using either of these parameters does not improve mAP score on the constructed multi-image dataset.

Example image that can be processed by the developed system is shown in Fig. 7. The image captures the whole plant. Overall, 346 detections of crucifier flea beetles and 64 instances of damaged leaf are found in this image. The image is fed into the model in resolution of 2016 x 2016.

Figure 7 – Example full plant image processed by the model

DOI 10.15588/1607-3274-2026-1-4

(a)



(b)

Figure 8 – Crops of full plant images, showing detections of pests only (a) and joint pests and leaf damage detection (b)

Crops of full plant images are shown in Fig. 8. In Fig. 8 (a) detection of crucifer flea beetles are shown; damaged leaf detections are filtered out for clarity. As can be seen, the vast majority of bugs have been detected. Only, blurry bug (top right) and bug in shadow region (middle left) have been missed. Obviously, these missed detections have no impact on the final decision. In Fig. 8 (b) joint damaged leaf and crucifer flea beetle detections are shown. Apparently, most of the bugs have left this region of the plant. However, the system has been able to detect severe damage taken by the plant.

CONCLUSIONS

In this work an important problem of cabbage pest detection at a distance has been solved. Existing datasets, such as IP102 and PlantDoc contain mostly macro photos of pests and plant diseases, which complicates training of neural networks for plant pest detection at a distance. To resolve the problem, additional cabbage photos at a distance have been collected in Dnipro region, Ukraine. YOLO 8, 9, 10, 11, RT-DETR neural networks have been analyzed. The best results have been shown by the YOLO 11 M (medium) network. Finally, augmentation hyperparameter search has been conducted using Tree-Structured Parzen estimator on multi-image validation set. The developed system requires only 7 annotated cabbage images to be trained. The final trained neural network can detect more than 300 instances of bugs on images of whole plants, even in cases when the bug is only 11×11 pixels on a 2016×2016 image. The developed system can be deployed for large-scale monitoring using edge devices. The system proposed in this work can detect large number of pests given a full plant image, which enables easier monitoring of large fields of cabbage.

The scientifical novelty of obtained results is that 1) Tree-Structured Parzen Estimator augmentation hyperparameter search on the proposed multi-image loss function on the validation has improved model performance on images at a distance from 0.3642 to 0.3892 mAP50–95; 2) cabbage pest and damaged leaf dataset has been collected.

The practical significance of obtained results is that the developed system is used for cabbage disease monitoring to prevent pest spread, and in contrast to previous systems detection is performed not on an individual leaf, but on a picture of a whole plant, which makes plant disease monitoring on large fields easier.

Prospects for further research are to propose custom object detection architecture for long distance plant health problem monitoring.

ACKNOWLEDGEMENTS

This research was carried out as part of the scientific project "Development of software and hardware of intelligent technologies for sustainable crop production in wartime and post-war" (state registration number 0124U000289) funded by the Ministry of Education and Science of Ukraine at the expense of the state budget.

DECLARATIONS

Conflict of interest: The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship, or otherwise, that could affect the research and its results presented in this paper.

Authors' contributions: Kostiantyn Khabarlak: software, data collection, writing – original draft preparation; Ivan Laktionov: writing – review and editing, funding acquisition; Vyacheslav Gorev: formal analysis, literature review; Grygorii Diachenko: method validation, writing – review and editing.

Data availability: The manuscript has no associated data.

Software availability: The manuscript has no associated software.

Use of artificial intelligence tools: The authors confirm that they did not use artificial intelligence technologies in creating the submitted work.

DOI 10.15588/1607-3274-2026-1-4

REFERENCES

1. Laktionov I. et al. A comprehensive review of recent approaches and Hardware-Software technologies for digitalisation and intellectualisation of Open-Field crop Production: Ukrainian case study in the global context, *Computers and Electronics in Agriculture,* 2024, Vol. 225, P. 109326. DOI: https://doi.org/10.1016/j.compag.2024.109326.
2. Gong H., Ma X., Guo Y. Research on a target detection algorithm for common pests based on an improved yolov7-tiny model, *Agronomy,* 2024, Vol. 14, № 12. DOI: 10.3390/agronomy14123068.
3. Teng Y. et al. MSR-RCNN: a multi-class crop pest detection network based on a multi-scale super-resolution feature enhancement module, *Frontiers in Plant Science*, 2022, Vol. 13. DOI: 10.3389/fpls.2022.810546.
4. Chakrabarty S. et al. Deep learning-based accurate detection of insects and damage in cruciferous crops using YOLOv5, *Smart Agricultural Technology,* 2024, Vol. 9, P. 100663. DOI: https://doi.org/10.1016/j.atech.2024.100663.
5. Dai M. et al. A new pest detection method based on improved YOLOv5M, *Insects,* 2023, Vol. 14, № 54. DOI: 10.3390/insects14010054.
6. Diachenko G. et al. An improved approach to prediction of maize disease occurrence based on weather monitoring and machine learning, *Case of the forest-steppe and northern steppe of Ukraine,* 2024, Vol. 12, № 4. DOI: 10.22364/BJMC.2024.12.4.03.
7. Bhargava A. et al. Plant leaf disease detection, classification, and diagnosis using computer vision and artificial intelligence: a review, *IEEE access : practical innovations, open solutions*, 2024, Vol. 12, pp. 37443–37469. DOI: 10.1109/ACCESS.2024.3373001.
8. Khabarlak K., Diachenko G., Laktionov I. Feature knowledge distillation using group convolutions for efficient plant pest recognition, *Proceedings of the 12th International Conference Information Control Systems & Technologies (ICST 2024).* Odesa, Ukraine, September 23–25 : CEUR workshop proceedings, CEUR-WS.org, 2024, Vol. 3790, pp. 377–387.
9. Girshick R.B. et al. Rich feature hierarchies for accurate object detection and semantic segmentation, *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*. Columbus, OH, USA, June 23–28, 2014, IEEE Computer Society, 2014, pp. 580–587. DOI: 10.1109/CVPR.2014.81.
10. Ren S. et al. Faster R-CNN: Towards real-time object detection with region proposal networks, *Advances in neural information processing systems 28: Annual conference on neural information processing systems 2015, December 7–12, 2015*. Montreal, Quebec, Canada, 2015, pp. 91–99.
11. Redmon J. et al. You only look once: Unified, real-time object detection, *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*. Las Vegas, NV, USA, June 27–30, 2016, IEEE Computer Society, 2016, pp. 779–788. DOI: 10.1109/CVPR.2016.91.
12. Bochkovskiy A., Wang C.-Y., Liao H.-Y.M. YOLOv4: Optimal speed and accuracy of object detection, *CoRR,* 2020, Vol. abs/2004.10934.
13. Jocher G. Ultralytics yolov5, 2020.
14. Carion N. et al. End-to-end object detection with transformers, *ECCV 2020, Glasgow, UK, August 23–28, 2020 : Lecture notes in computer scienc*e. Springer, 2020, Vol. 12346. pp. 213–229. DOI: 10.1007/978-3-030-58452-8_13.
15. Wu X. et al. IP102: A large-scale benchmark dataset for insect pest recognition, *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, IEEE, 2019. DOI: 10.1109/cvpr.2019.00899.
16. Li A. et al. D3-YOLOv10: Improved YOLOv10-based lightweight tomato detection algorithm under facility scenario, *Agriculture* (Nitra, Slovakia), 2024, Vol. 14, № 12. DOI: 10.3390/agriculture14122268.
17. Khabarlak K. S. Faster optimization-based meta-learning adaptation phase, *Radio Electronics, Computer Science, Control,* 2022, № 1, P. 82. DOI: 10.15588/1607-3274-2022-1-10.
18. Nuthalapati S. V., Tunga A. Multi-domain few-shot learning and dataset for agricultural applications, *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021*. Montreal, QC, Canada, October 11–17, 2021, IEEE, 2021, pp. 1399–1408. DOI: 10.1109/ICCVW54120.2021.00161.
19. Wang X. et al. Prior knowledge auxiliary for few-shot pest detection in the wild, *Frontiers in Plant Science*, 2023, Vol. 13–2022. DOI: 10.3389/fpls.2022.1033544.
20. Jocher G., Chaurasia A., Qiu J. Ultralytics YOLOv8, 2023.
21. Wang C.-Y., Liao H.-Y.M. YOLOv9: Learning what you want to learn using programmable gradient information, 2024.
22. Wang A. et al. YOLOv10: Real-time end-to-end object detection, *CoRR,* 2024, Vol. abs/2405.14458. DOI: 10.48550/ARXIV.2405.14458.
23. Jocher G., Qiu J. Ultralytics YOLO11, 2024.
24. Zhao Y. et al. DETRs beat YOLOs on real-time object detection, *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024*. Seattle, WA, USA, June 16–22, 2024, IEEE, 2024, pp. 16965–16974. DOI: 10.1109/CVPR52733.2024.01605.
25. Singh D. et al. PlantDoc: A dataset for visual plant disease detection, *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*. New York, NY, USA, Association for Computing Machinery, 2020, pp. 249–253. DOI: 10.1145/3371158.3371196.
26. Cubuk E. D. et al. AutoAugment: Learning augmentation policies from data, *CoRR,* 2018, Vol. abs/1805.09501.
27. Bergstra J. et al. Algorithms for hyper-parameter optimization, *Advances in neural information processing systems 24: 25th annual conference on neural information processing systems 2011, Proceedings of a meeting held 12–14 December 2011*. Granada, Spain, 2011, pp. 2546–2554.
28. Akiba T. et al. Optuna: a next-generation hyperparameter optimization framework, *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery and data mining,* 2019.

Received 01.08.2025.
Accepted 15.01.2026.
Published 27.03.2026.

DOI 10.15588/1607-3274-2026-1-4

УДК 004.93

МЕТОД ВИЯВЛЕННЯ ПОШКОДЖЕНЬ ТА ШКІДНИКІВ КАПУСТИ З ДАЛЕКОЇ ВІДСТАНІ НА ОСНОВІ YOLO11

Хабарлак К. С. – д-р філософії, доцент кафедри системного аналізу та управління, Національний технічний університет «Дніпровська політехніка», Дніпро, Україна. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0003-4263-0871.

Лактіонов І. С. – д-р техн. наук, професор, професор кафедри програмного забезпечення комп'ютерних систем, Національний технічний університет «Дніпровська політехніка», Дніпро, Україна. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0001-7857-6382.

Горєв В. М. – канд. фіз.-мат. наук, доцент, завідувач кафедри фізики, Національний технічний університет «Дніпровська політехніка», Дніпро, Україна. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0002-9528-9497.

Дяченко Г. Г. – канд. техн. наук, доцент кафедри електропривода, Національний технічний університет «Дніпровська політехніка», Дніпро, Україна. ROR: https://ror.org/05hkn5555. ORCID: https://orcid.org/0000-0001-9105-1951.

АНОТАЦІЯ

Актуальність. Для забезпечення стабільного врожаю необхідно постійно контролювати стан рослин і вчасно вживати заходів для запобігання поширенню захворювань. Традиційні підходи базуються на ручному огляді рослин, в той же час для навчання нейронних мереж потрібні великі обсяги анотованих даних. Як ручний огляд, так і анотування даних вимагають експертних знань і потребують багато часу. Для навчання часто використовуються фотографії листя з близької відстані, оскільки їх легше знайти в Інтернеті. Однак це ускладнює оцінку поширення хвороб на великих ділянках. Капуста є однією з рослин, що широко вирощуються в Україні, але існує недостатня кількість досліджень, присвячені моніторингу здоров'я капусти.

Мета. Метою цієї роботи є створення системи виявлення хвороб і шкідників капусти на основі нейронної мережі, яку можна навчити на невеликій кількості навчальних зображень. При виведенні система повинна виявляти шкідників на зображеннях рослин на відстані цілої рослини.

Метод. З огляду на те, що існуючі набори даних про хвороби рослин, такі як IP102 і PlantDoc, містять переважно знімки хворих рослин зблизька, мережі, навчені на таких наборах даних, страждають від відсутності узагальнення до зображень на відстані. Для вибору найкращої моделі виявлення об'єктів у роботі було проаналізовано найсучасніші архітектури виявлення об'єктів, а саме YOLO 8, 9, 10, 11 і RT-DETR. Для збільшення відстані виявлення запропоновано функцію втрат на декількох зображеннях для поліпшення пошуку гіперпараметрів на основі методу Tree-Structured Parzen Estimators (TPE). Крім того, для поліпшення якості виявлення було зібрано новий набір даних хвороб капусти в Дніпропетровській області України. Нові класи включають хрестоцвітну блішку (поширений шкідник у Дніпропетровській області) та пошкоджене листя. Коли шкідника не видно, але пошкодження листя є, визначити конкретного шкідника може бути неможливо. Тому ми вводимо додатковий клас пошкодженого листя, який фіксує загальне пошкодження рослин. Це також дозволяє відстежувати швидкість одужання рослин, коли вжито заходів для зупинення поширення шкідників. Ми поєднуємо зібрані зображення з більшим набором даних IP102, щоб збільшити кількість охоплених шкідників і сформувати новий набір даних Cabbage+IP102.

Результати. 1) Пошук за допомогою TPE, використовуючи функцію втрат на кількох зображеннях, покращив YOLO 11 M з 0,3642 до 0,3892 mAP50–95 на зображеннях, зроблених на відстані. 2) Зібраний набір даних дозволив виявляти проблеми зі здоров'ям капустяних рослин на відстані, включаючи випадки, коли шкідника наразі не видно, але пошкодження є.

Висновки. У цій роботі представлено систему виявлення шкідників капусти та пошкодженого листя на основі YOLO 11 M. Архітектура детектора була обрана як найкраща під час аналізу 2 наборів даних. Розроблена система вимагає лише 7 анотованих зображень капусти для навчання та виявлення шкідників і пошкодженого листя на зображеннях високої роздільної здатності (2016x2016) цілих рослин капусти. Кінцева модель може бути використана для моніторингу проблем зі здоров'ям капусти, пошкоджень та швидкості загоєння за допомогою зображень, зроблених на відстані.

КЛЮЧОВІ СЛОВА: сільське господарство, глибоке навчання, моніторинг здоров'я рослин, виявлення шкідників, оцінка пошкодження листя, YOLO 11, родина капустяні.

ЛІТЕРАТУРА

1. A comprehensive review of recent approaches and Hardware-Software technologies for digitalisation and intellectualisation of Open-Field crop Production: Ukrainian case study in the global context / I. Laktionov [et al.] // Computers and Electronics in Agriculture. – 2024. – Vol. 225. – P. 109326. – DOI: https://doi.org/10.1016/j.compag.2024.109326.
2. Gong H. Research on a target detection algorithm for common pests based on an improved yolov7-tiny model / H. Gong, X. Ma, Y. Guo // Agronomy. – 2024. – Vol. 14, № 12. – DOI: 10.3390/agronomy14123068.
3. MSR-RCNN: a multi-class crop pest detection network based on a multi-scale super-resolution feature enhancement module / Y. Teng [et al.] // Frontiers in Plant Science. – 2022. – Vol. 13. – DOI: 10.3389/fpls.2022.810546.
4. Deep learning-based accurate detection of insects and damage in cruciferous crops using YOLOv5 / S. Chakrabarty [et al.] // Smart Agricultural Technology. – 2024. – Vol. 9. – P. 100663. – DOI: https://doi.org/10.1016/j.atech.2024.100663.
5. A new pest detection method based on improved YOLOv5M / M. Dai [et al.] // Insects. – 2023. – Vol. 14. – № 54. – DOI: 10.3390/insects14010054.
6. An improved approach to prediction of maize disease occurrence based on weather monitoring and machine learning / G. Diachenko [et al.] // Case of the forest-steppe and northern steppe of Ukraine. – 2024. – Vol. 12, № 4. – DOI: 10.22364/BJMC.2024.12.4.03.

DOI 10.15588/1607-3274-2026-1-4

7. Plant leaf disease detection, classification, and diagnosis using computer vision and artificial intelligence: a review / A. Bhargava [et al.] // IEEE access : practical innovations, open solutions. – 2024. – Vol. 12. – P. 37443–37469. – DOI: 10.1109/ACCESS.2024.3373001.
8. Khabarlak K. Feature knowledge distillation using group convolutions for efficient plant pest recognition / K. Khabarlak, G. Diachenko, I. Laktionov // Proceedings of the 12th International Conference Information Control Systems & Technologies (ICST 2024), Odesa, Ukraine, September 23–25 : CEUR workshop proceedings. – CEUR-WS.org, 2024. – Vol. 3790. – P. 377–387.
9. Rich feature hierarchies for accurate object detection and semantic segmentation / R. B. Girshick [et al.] // 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23–28, 2014. – IEEE Computer Society, 2014. – P. 580–587. – DOI: 10.1109/CVPR.2014.81.
10. Faster R-CNN: Towards real-time object detection with region proposal networks / S. Ren [et al.] // Advances in neural information processing systems 28: Annual conference on neural information processing systems 2015, December 7–12, 2015, Montreal, Quebec, Canada. – 2015. – P. 91–99.
11. You only look once: Unified, real-time object detection / J. Redmon [et al.] // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016. – IEEE Computer Society, 2016. – P. 779–788. – DOI: 10.1109/CVPR.2016.91.
12. Bochkovskiy A. YOLOv4: Optimal speed and accuracy of object detection / A. Bochkovskiy, C.-Y. Wang, H.-Y.M. Liao // CoRR. – 2020. – Vol. abs/2004.10934.
13. Jocher G. Ultralytics yolov5 / G. Jocher. – 2020.
14. End-to-end object detection with transformers / N. Carion [et al.] // ECCV 2020, Glasgow, UK, August 23–28, 2020 : Lecture notes in computer science. – Springer, 2020. – Vol. 12346. – P. 213–229. – DOI: 10.1007/978-3-030-58452-8_13.
15. IP102: A large-scale benchmark dataset for insect pest recognition / X. Wu [et al.] // 2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR). – IEEE, 2019. – DOI: 10.1109/cvpr.2019.00899.
16. D3-YOLOv10: Improved YOLOv10-based lightweight tomato detection algorithm under facility scenario / A. Li [et al.] // Agriculture (Nitra, Slovakia). – 2024. – Vol. 14, № 12. – DOI: 10.3390/agriculture14122268.
17. Khabarlak K.S. Faster optimization-based meta-learning adaptation phase / K. S. Khabarlak // Radio Electronics, Computer Science, Control. – 2022. – № 1. – P. 82. – DOI: 10.15588/1607-3274-2022-1-10.
18. Nuthalapati S.V. Multi-domain few-shot learning and dataset for agricultural applications / S. V. Nuthalapati, A. Tunga // IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, QC, Canada, October 11–17, 2021. – IEEE, 2021. – P. 1399–1408. – DOI: 10.1109/ICCVW54120.2021.00161.
19. Prior knowledge auxiliary for few-shot pest detection in the wild / X. Wang [et al.] // Frontiers in Plant Science. – 2023. – Vol. 13–2022. – DOI: 10.3389/fpls.2022.1033544.
20. Jocher G. Ultralytics YOLOv8 / G. Jocher, A. Chaurasia, J. Qiu. – 2023.
21. Wang C.-Y. YOLOv9: Learning what you want to learn using programmable gradient information / C.-Y. Wang, H.-Y.M. Liao. – 2024.
22. YOLOv10: Real-time end-to-end object detection / A. Wang [et al.] // CoRR. – 2024. – Vol. abs/2405.14458. – DOI: 10.48550/ARXIV.2405.14458.
23. Jocher G. Ultralytics YOLO11 / G. Jocher, J. Qiu. – 2024.
24. DETRs beat YOLOs on real-time object detection / Y. Zhao [et al.] // IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16–22, 2024. – IEEE, 2024. – P. 16965–16974. – DOI: 10.1109/CVPR52733.2024.01605.
25. PlantDoc: A dataset for visual plant disease detection / D. Singh [et al.] // Proceedings of the 7th ACM IKDD CoDS and 25th COMAD. – New York, NY, USA: Association for Computing Machinery, 2020. – P. 249–253. DOI: 10.1145/3371158.3371196.
26. AutoAugment: Learning augmentation policies from data / E. D. Cubuk [et al.] // CoRR. – 2018. – Vol. abs/1805.09501.
27. Algorithms for hyper-parameter optimization / J. Bergstra [et al.] // Advances in neural information processing systems 24: 25th annual conference on neural information processing systems 2011. Proceedings of a meeting held 12–14 December 2011, Granada, Spain. – 2011. – P. 2546–2554.
28. Optuna: a next-generation hyperparameter optimization framework / T. Akiba [et al.] // Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery and data mining. – 2019.

DOI 10.15588/1607-3274-2026-1-4