Article:

Li, S., Yuan, Z., Peng, R. et al. (4 more authors) (2024) An effective farmer-centred mobile intelligence solution using lightweight deep learning for integrated wheat pest management. Journal of Industrial Information Integration, 42. 100705. ISSN 2467-964X

https://doi.org/10.1016/j.jii.2024.100705

An Effective Farmer-Centred Mobile Intelligence Solution Using Lightweight Deep Learning for Integrated Wheat Pest Management

ARTICLE INFO

Keywords:
Integrated Pest Management
Deep Learning
Expert System
Smart Agriculture
Tiny Object Detection

ABSTRACT

Integrated Pest Management (IPM) techniques have been widely used in agriculture to manage pest damage in the most economical way and to minimise harm to people, property and the environment. However, current research and products on the market cannot consolidate this process. Most existing solutions either require experts to visually identify pests or cannot automatically assess pest levels and make decisions based on detection results. To make the process from pest identification to pest management decision making more automated and intelligent, we propose an end-to-end integrated pest management solution that uses deep learning for semi-automated pest detection and an expert system for pest management decision making. Specifically, a low computational cost sampling point generation algorithm is proposed to enable mobile devices to generate uniformly distributed sampling points in irregularly shaped fields. We build a pest detection model based on YoloX and use Pytorch Mobile to deploy it on mobile phones, allowing users to detect pests offline. We develop a standardised sampling specification and a mobile application to guide users to take photos that allow pest population density to be calculated. A rule-based expert system is established to derive pest management thresholds from prior agricultural knowledge and make decisions based on pest detection results. We also propose a human-in-the-loop algorithm to continuously track and update the validity of the thresholds in the expert system. The mean average precision of the pest detection model is 58.17% for 97 classes, 75.29% for 2 classes, and 57.33% for 11 classes on three pest datasets, respectively. The usability of the pest management system is assessed by the User Experience Surveys and achieves a System Usability Scale (SUS) score of 76. The usability of the proposed solution is validated by qualitative field experiments.

1. Introduction

Wheat is an important food crop and is considered one of the world's four major food crops, along with rice, maize and potatoes. Wheat is used as a staple food in more than 100 countries worldwide (Curtis, Rajaram, Gómez Macpherson et al., 2002). About one third of the world's population depends on wheat as a staple food and it accounts for 27% of global cereal production (Shewry, 2009). In UK, it is estimated that estimate that wheat comprises 25% of the daily calorific intake (Mottaleb, Kruseman and Snapp, 2022).

The loss in potential yield from pest attack i.e., insect and mollusc, can be substantial, to the point of total loss of crop. Recently investigators have examined the effects of pest attack on wheat yield. The Food and Agriculture Organisation of the United Nations (FAO) estimates that between 20% and 40% of global crop production is lost to pests each year (Department for Environment, Food & Rural Affairs, 2020). Plant diseases cost the global economy an estimated $220 billion annually, while invasive insects cost an estimated $70 billion (Sarkozi, 2019).

Studies by Nacarrow et al. and Dedryver et al. suggest up to 80% yield loss due to virus transmission by aphids and 5-20% yield loss due to direct feeding damage (Nancarrow, Aftab, Hollaway, Rodoni and Trębicki, 2021; Dedryver, Ralec and Fabre, 2010). Orange wheat blossom midges are native to Europe and have spread to major wheat-producing countries around the world (Senevirathna, Guelly and Mori, 2023). Currently, both aphids and orange wheat blossom midges are damaging pests of wheat in the UK (Ellis, White, Holland, Smith, Collier and Jukes, 2014). Due to the low cost of insecticides, the economic return from additional producion is six times the cost of treating aphids (Redman, 2022).

Hence growers apply pesticides to mitigate potential yield loss These applications are often done on an insurance basis (i.e., an application is made as a contingency to mitigate potential yield loss) because pest abundance is high. These applications are potentially wasteful (no economic benefit) and damaging to the environment. With sustainable crop protection becoming more important, there is increasing demand for decision support systems that can help farmers grow crops more sustainably with fewer chemical interventions.

To help address these issues, a large and growing body of literature has investigated the pest identification and the economic threshold levels. In addition to the population density of insects or the extent of crop damage, economic thresholds used for integrated pest management often contain relevant contexutal information, such as climatic, geographic, and phenological information. For example, the resistance to pests increases if the crop reaches a late growth stage; pesticide applications are usually not recommended on rainy days because the effect is weakened, etc. The Agriculture and Horticulture Development Board (AHDB) have produced an encyclopaedia of pests and natural enemies in field crops. This provides all the information required to make an informed decision on whether pest control is warranted or not (Agriculture and Horticulture Development Board, 2022). Although the reference manual is very comprehensive, it is not specific to wheat and not very user friendly in a field situation either as a hard document or on a mobile

ORCID(s):

phone. A new tolerance-based decision support system to minimise the risk of crop damage by wheat bulb fly (WBF) has been devised under IPM principles by ADAS, a UK-based independent agricultural and environmental consultancy (Leybourne, Storer, Berry and Ellis, 2022). However, the identification of the pest and a risk-based decision still needs to be made by agronomists with specialist knowledge. To automate the detection of pest species, artificial intelligence scientists are using objective detection algorithms for pest identification. Nevertheless, these deep learning-based algorithms can only identify the type of pest, but cannot quantify the severity of the current pest. There are two main scientific problems that contribute to this issue. First, current pest thresholds in the agricultural literature are difficult to use in computer vision, for example, some pest thresholds are measured in terms of the number of pests per plant, yet it is difficult for deep learning models to distinguish between different plants. Secondly, because the actual area of a photograph is not known, the density of the pest population in the photograph cannot be calculated, so it is not possible to measure the severity of the infestation directly from the photograph. In addition, the economic thresholds for wheat vary according to climate, water and heat conditions and pest species, and sometimes pests develop resistance, making it difficult to use a constant set of pest thresholds for decision making in all environments.

This study has proposed a solution of integrated pest management decision making for wheat pest aims to the research problems mentioned above. The system combines deep learning models for pest detection and counting with an expert system for pest management decisions, with specific contributions including:

- to design and train a light-weight deep learning model for semi-automatic wheat pest detection on smartphones.
- to propose a sampling standard and a computational graphics-based algorithm for sampling point generation that reflects the challenges of quantifying pest severity from deep learning pest detection results.
- to convert the text-based thresholds for wheat pests in the literature into a rule-based expert system to overcome the difficulties of using textual prior knowledge for computer vision-based integrated pest management.
- to implement a human-in-the-loop threshold optimisation algorithm to semi-automatically adjust inaccurate thresholds due to pesticide resistance or regional differences.

The remainder of the paper is structured as follows. Section 2 reviews the state of the art research on object detection and integrated pest management. Section 3 presents the datasets and the proposed solution of the semi-automatic integrated pest management decision making system. Section 4 evaluates the performance of the deep learning based pest detection model and the usability of the proposed pest management decision making system. Section 5 briefly concludes the proposed approaches presented in section 3 along with an outline of future work.

2. Literature Review

The scope of this research is deep learning based pest identification and expert system based decision making for pest management. Therefore, the literature review in this section is divided into two parts, the first providing an overview of relevant deep learning techniques in the literature for target detection and the second outlining the application of expert systems in agriculture.

2.1. Object Detection

Object detection is one of the important tasks in computer vision to identify and localise all instances of object in the image data. Early work on object detection was based on hand-crafted feature extractors, such as the histogram of oriented gradients (Dalal and Triggs, 2005) and Harris corner detector (Harris, Stephens et al., 1988). However, for complex multi-classification object detection tasks, these traditional methods lose their effectiveness.

The convolutional neural networks (CNNs) were proposed to solve the problem of low performance of hand-craft features by automatically exploring effective features using large amounts of image data, such as VGG (Simonyan and Zisserman, 2014), ResNet (He, Zhang, Ren and Sun, 2016), and CSPNet (Wang, Liao, Wu, Chen, Hsieh and Yeh, 2020a). Based on the superiority of convolutional neural networks, a series of deep learning-based object detection models have been proposed, which is divided into two-stages detectors and one-stages detectors. The two-stages detector divides the detection process into two steps, the regional proposal stage, and the detection stage. In contrast, the one-stages detector proposed bounding box and classified object in one stage. From the view of model structure, the difference between the two-stage detector and one-stage detector lies in the presence or absence of a separate module for generating bounding box.

Faster Region-based Convolutional Neural Network (Faster RCNN) (Ren, He, Girshick and Sun, 2015) is the latest work following the design of RCNN (Girshick, Donahue, Darrell and Malik, 2014) detection model family, which are all two-stage detection models. As the definition of the two-stage detection model, the models structure of RCNN family can be divided into two steps, the region of interest proposal stage and detection stage. In the early RCNN (Girshick et al., 2014), a traditional algorithm Selective Search (Uijlings, Van De Sande, Gevers and Smeulders, 2013) was used to propose 2000 regions of interest. The proposed regions were then warped and propagated through a CNN backbone. The final detection results were subsequently obtained by Support Vector Machines (SVMs) and Non-maximum suppression (NMS). In order to increase the speed of detection, Faster RCNN use a CNN as a region proposal network (RPN) to propose regions of interest with associated objectness score. The multi-scale bounding boxes obtained by RPN were combined with the feature maps in the backbone network and passed through a classifier and bounding box regressor to obtain the detection results.

In contrast, the Yolo detection model (Ge, Liu, Wang, Li and Sun, 2021) family is representative of the one-stage detectors, which solve the detection problem by directly predicting the likelihood of related pixels being a detection object and the bounding box properties in one stage. This approach used convolutional neural networks to separate the original input images into grids and predict the bounding boxes and object scores for each grid, allowing for a simpler and smaller model to detection. Those models gained faster detection at the cost of detection accuracy in the early works. In recent work of YoloX (Ge et al., 2021), this cost is offset by a large number training tricks and the adaptation of the model structure. Specifically, various data augmentation methods, batch normalisation, and CLoU loss function were used in the training phase of the detection model. In terms of model structure, Cross-stage partial connections, SPP-Block, PAN path aggregated block neck, Decoupling detection head were used to optimise the model structure to achieve fast and accurate detection. Overall, one-stage detection model solves the problem of fast and accurate object detection in a simpler way.

2.2. Expert Systems

Expert systems use computer models derived from human experts to deal with complex real-world problems that require expert interpretation, and reach the same results as experts (Liao, 2005). The Agricultural Expert System (AES) applies expert system technology to the agricultural sector. It summarises and brings together knowledge and techniques from the field of agriculture and the knowledge of agricultural experts, as well as data obtained through experiments and mathematical models to simulate the decision-making process of agricultural experts.

Since the 1980s, specialist systems technology has been applied to agricultural problems, particularly in the area of integrated pest management, which has been in development for a relatively long time and is particularly well developed (Gerevini, Perini, Ricci, Forti, Ioriatti, Mattedi, Monetti et al., 1992; El-Azhary, Hassan and Rafea, 2000; Harrison, 1991). S. Kaloudis et al. describe an expert system for the identification of forest pests and the provision of related control measures. The system identifies more than 40 species of forest pests based on their growth stage, the damage caused by the pests and the results of their research in the forest. Once a pest has been identified, the system will provide a suitable treatment plan to minimise damage to the forest by the pest (Kaloudis, Anastopoulos, Yialouris, Lorentzos and Sideridis, 2005). CUPTEX is an expert system that has been developed to manage cucumber pests and diseases. The main purpose of the system is to identify the causes of anomalies and to make appropriate treatment recommendations. In this case, the system starts with the identification of the cause before recommendations are given (Rafea, El-Azhari, Ibrahim, Edres, Mahmoud and Street, 1995). The Tomato Expert System developed by Yialouris and Siderdis was used to deal with the problem of identifying tomato pests and diseases. A framework knowledge representation table was used to describe the knowledge base, and notably fuzzy logic was used to deal with uncertainty in the diagnosis (Yialouris and Sideridis, 1996).

3. Materials and methods

This work aims to automate the process of integrated pest management for wheat. To automate pest detection, we introduce deep learning, which relies on a large amount of data. To address this research question, we perform data augmentation of the collected data. Another research problem that hinders the automation of integrated pest management is the interaction between deep learning model detection results and a decision-making expert system. To address this challenge, we propose a sample point generation algorithm to aid sampling and a density calculation algorithm to quantify the pest detection results so that they can be used in an expert system. This section also concludes with a description of the human-in-the-loop algorithm for automatic correction of pest thresholds in expert systems

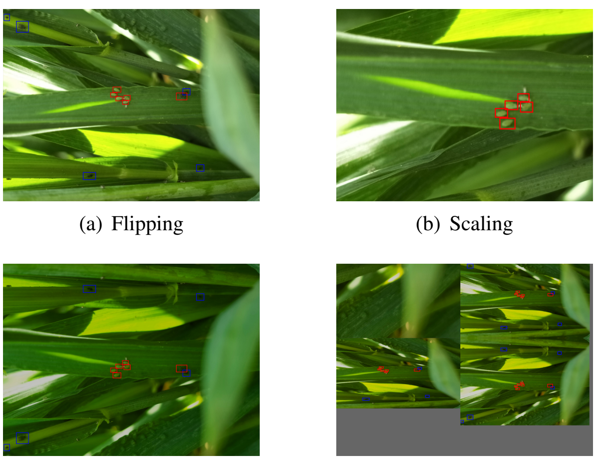
3.1. Pest Datasets

Multiple pest datasets are used for the validation of pest detection models, including both public and private datasets. IP102 (Wu, Zhan, Lai, Cheng and Yang, 2019) is a public dataset that includes 19 thousands pest images with annotation belonging to 102 classes and 51 thousands pest images without annotation. The images in the IP102 are collected through a search engine, so the backgrounds are more diverse. In addition, the images in the IP102 have a larger percentage of pests than that images collected in real environments. In comparison, the AgriPest dataset (Wang, Liu, Xie, Yang, Li and Zhou, 2021b) includes 49.7k pest images of 14 species collected from a natural environment with fixed equipment and mobile equipment. We select a subset of the AgriPest dataset containing two types of aphids by manual screening to verify the ability of the detection model in a realistic sampling scenario. In addition, we collected image data using mobile equipment on three different UK farms according to the proposed sampling specifications. These three datasets show the different challenges that the pest detection task poses to object detection models. Firstly, IP102 (Wang et al., 2021b) and our datasets contain a large number of insect species, which challenges the classification ability of object detection models. Secondly, datasets collected in real environments, such as the AgriPest dataset (Wang et al., 2021b) and our dataset, face the challenge of tiny object detection. Last, all three datasets suffer from data imbalance and limited dataset size.

For improving the accuracy of the detection model, multiple data augmentation methods are used during the model training phase. The data augmentation methods include basic image transformations, such as random flipping, random scaling, and random HSV colour perturbation. In

Table 1
Statistical information of datasets. The columns in the table show the total number of samples, the total number of classifications, the number of the largest category, the number of the smallest category, and the average percentage of one object pixels in the image.

	IP102	AgriPest	Our Dataset
Num. of samples	19,167	1,000	4,270
Num. of objects	22,284	6,325	8,303
Num. of classifications	97	2	11
Max. Num. of a category	2,975	4,755	5,575
Min. Num. of a category	2	1,570	3
Avg. object pixels pct.	37.27	0.08	0.13



(a) Flipping (b) Scaling

(c) HSV perturbation (d) Mosaic

Figure 1: Data Augmentation. The flipping method randomly flips the original image horizontally or vertically. The scaling method randomly scales the original image and fills the border with grey. The HSV perturbation randomly adjust saturation, hue, and lightness. The mosaic randomly selects and slices four transferred images.

the work of YoloX, Mosaic (Ge et al., 2021) is proposed for improving the model accuracy, which splices four images randomly after basic image transformations. The augmented image data is shown in Figure 1. This artificially constructed training data contains more invariance and enriches the training sample to improve the accuracy of the model.

3.2. Integrated pest management decision making system

Automatic in-field pest detection and recognition using mobile vision technique is a hot topic in modern intelligent agriculture but suffers from serious challenges including complexity of wild environment, detection of tiny size pest and classification of multiple classes of pests. To overcome these obstacles, the popular methods are to design a Convolutional Neural Network (CNN) model that extracts visual features and identifies crop disease images based on these features. These methods work well on laboratory environment under simple background but achieve low accuracy and poor robustness in processing the raw images captured from practical fields that contain inevitable noises. Motivated by the above mentioned inadequacy of existing studies, a light-weight deep learning model for automatic wheat pest detection architecture is established to fuse the features of pest images and the features of contextual information to be deployed on mobile devices towards pest recognition and detection in the wild and make decisions of pest treatments.

The proposed architecture consists of three parts: server, interface and local library. The server refers to a kubernetes cluster that manages a number of RESTful web services for user management, farm management, pest encyclopedia, decision making, thresholds optimisation function. The interface and local library are implemented by Kotlin for Android device.

Fig. 2 also displays an overall process of users to use the system. Prior to using the system, users login the logs in on the mobile application and the server grants access to the successfully logged-in user. After logging in, the application requests the server to obtain the field information associated with the current user. Then the user selects the field for pest management and selects the growth stage of the current crop. At the same time, the sampling point generation algorithm in the local library generates sampling points for the selected field. Then the application interface jumps to the map interface of the selected field, which shows the generated sampling points and the user's location, and the user goes to each sampling point in turn to take pictures. Each sampled picture calls the pest detection model in the local library for classification and counting, and calls the density calculation model to calculate the population density of the pests detected in the photo. When all sampling points are sampled, the pest detection results and population density calculation results will be demonstrated to the users. Users are able to manually modify, add, delete the detection and calculations results. The results are uploaded to the decision making expert system on the server to request pest management recommendations after user confirmation of the results.

In the pest management suggestion interface, the application also requests the description of detected pests from the Pest encyclopedia server. Every time a pest management decision is completed, the system will send a questionnaire to the user to evaluate the effect of the last pest detection, and the user's feedback will be returned to the threshold optimisation algorithm in the server to optimise decision-making expert system.

3.2.1. Pest Detection Model

In this study, we address the technical challenge of automatically estimating pest population densities through object detection model. (Yuan, Li, Yang and Li, 2022) As described in related work, object detection models provide the ability to identify a bounding box with classification for each object of interest in an input image. We are inspired by the Yolo detection models, which are lightweight and effective object detection models, to propose a pest detection model. The

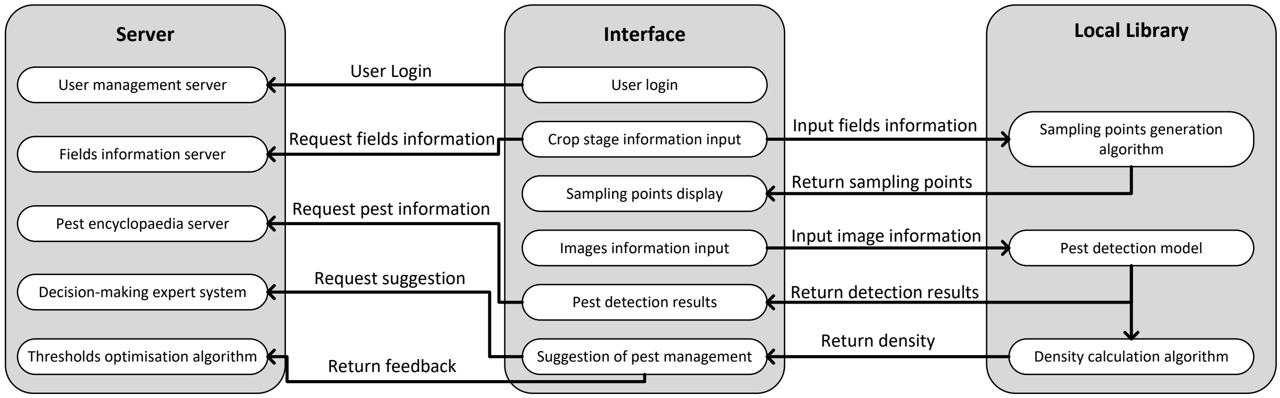


Figure 2: Interaction between server, interface and local library

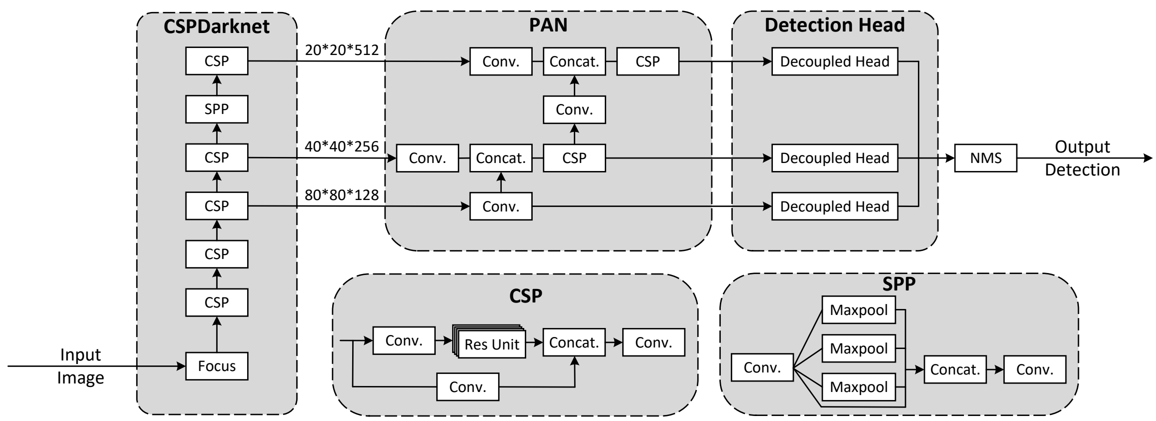


Figure 3: The detection model structure. The CSP and SPP are submodels consisting of convolutional layers (Conv.), concatenate layers (Concat.), and max pooling layers (Maxpool) in CSPDarknet (Ge et al., 2021).

architecture of the proposed detection model is shown in Figure 3, including a CNN backbone for features extraction, a detection neck for fusion of multi-layer features, multiple decoupled detection heads for obtaining the potential bounding box and corresponding classification information in the input image, and a Non-extreme suppression for obtaining the final detection result.

In our detection model, we use CSPDarknet (Bochkovskiy, Wang and Liao, 2020) as the backbone. In the CSPDarknet, each CSP module has a residual block to learn more and different features, which facilitates the accuracy of small object detection. In addition, Spatial Pyramid Pooling is used before the last CSP module to improve the perceptual field of the network by pooling with different size of maximum pooling kernels. An improved version of the ReLU activation function, SiLU (Elfwing, Uchibe and Doya, 2018), is used throughout the detection model, which has a smoother gradient change compared to the original ReLU activation function. For detection neck, we use Path Aggregate Network (Liu, Qi, Qin, Shi and Jia, 2018) which is more accurate in tiny object detection. The decoupled detection heads used separate convolutional neural networks for classification, bounding box, and object score prediction, improving detection accuracy at the cost of an acceptable number of parameters.

3.2.2. *Generating Evenly Distributed Sampling Points*

Generating evenly distributed sampling points is the first step in pest management. There are many mature sampling point selection methods in the agricultural field. Such as five-point sampling method, equidistant sampling method, grid sampling method, etc. However, these methods need to be used manually by a person. When we use computers to generate sample points using these methods, it is not guaranteed that all the points generated will be in the field because the computer cannot tell if a point is inside or outside the field (see figure 4(a)(b)(c)(d)(e)(f). This is not usually a problem in areas with large plains. However, it can limit the use of our software in areas with complex field shapes.

To overcome the dependency of the agricultural experts on sample point selection, computer science researchers started to develop computer-aided sample point selection methods. A representative method for selecting uniform sampling points is developed by ArcGIS and is based on computational graphics. The mathematical basis of the

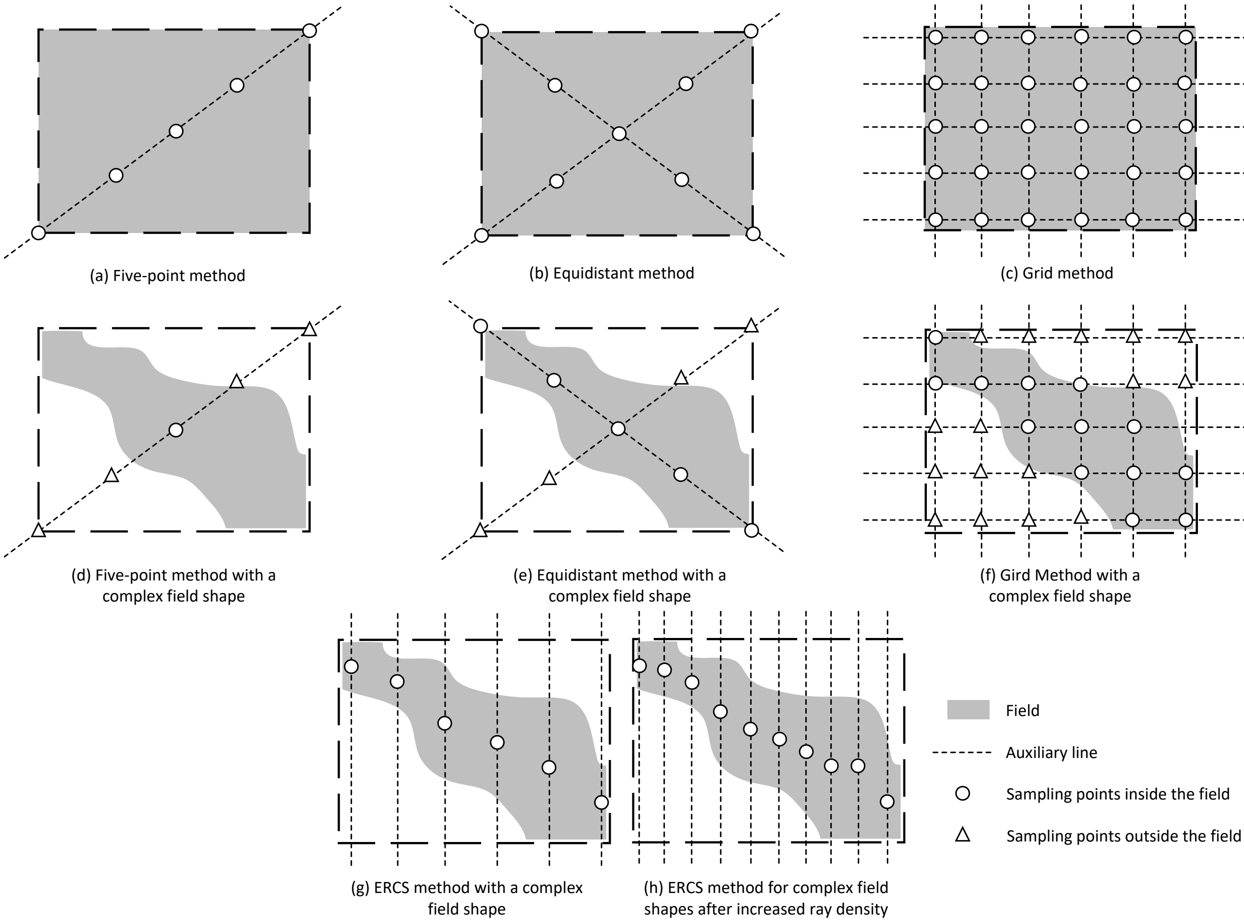


Figure 4: Comparison of conventional sampling point generation methods with the proposed method: Equidistant Ray Casting Sampling (ERCS)

method is triangulation. This method can generate very uniform sampling points, but its computational cost is extremely high, and it needs to generate a large number of sampling points to make sure these points are uniformly distributed which will significantly increase the workload at our user end.

In response to the disadvantages of both traditional methods, modern methods, and computer-aid methods, we proposed our own methods which can generate relatively uniform sampling points with exceptionally low computational cost and the number of sampling points is significantly reduce to relief our users from heavy workload. In principle, our approach is based on two theories: equidistant sampling method and ray casting algorithm. Equidistant sampling is also known as equal-distance sampling which is been widely used by the agronomists. Equidistant sampling first divides the sampled field into several equal parts, the distance or interval is determined by the sampling ratio, and then the sample squares are drawn according to this equal distance or interval in order to get uniformly distributed sampling points. To addressing the challenge of determining whether the generated sampling points are within the polygonal fields, we introduced ray casting algorithm. This algorithm is sometimes also known as the crossing number algorithm or the even–odd rule algorithm, and was known as early as 1962 (Shimrat, 1962). The algorithm is based on a simple observation that if a point moves along a ray from infinity to the probe point and if it crosses the boundary of a polygon, possibly several times, then it alternately goes from the outside to inside, then from the inside to the outside, etc. As a result, after every two "border crossings" the moving point goes outside. This observation may be mathematically proved using the Jordan curve theorem.

By fusing these two method and algorithm, we proposed our sampling methods: Equidistant Ray Casting Sampling (ERCS). ERCS firstly first place the field in a rectangle, the size of which depends on the coordinates of the point at the very edge of the field. Rays then vertically and equally divide the rectangle. According to the ray casting algorithm, the computer will be able to know which part of the ray is inside the polygons by counting the number of intersections between the ray and the field's boundaries. Hence, the midpoints of the line segment inside the polygon will be selected as the sampling points. In addition, as shown in figure 4(g)(h) by adjusting the distances between the rays, our users can adjust the number of sampling points, making it easy to optimise their workloads.

3.2.3. Calculating population densities of pests using single photographs

At present, most of the products on the market only do the previous step, that is, pest detection. However, in order to realise semi-automatic IPM in the whole process, we not only need to realise pest detection, but also need to conduct quantitative analysis on the detection results. In order to achieve this goal, we need to relate the number and species of pests detected by the deep learning model with our prior knowledge of agriculture (Economic thresholds for integrated pest management). However, the current existing thresholds are usually the population density per unit area or the number of pests per crop, whereas deep learning models can only detect the species and quantity of pests in a photo and cannot calculate the population density of each type of pest, as the actual area of the photo is unknown. It is also difficult for deep learning models to detect the type and number of pests on a single plant, because when taking pictures of most densely planted crops, one photo usually contains multiple plants.

To achieve a link between thresholds in a prior agricultural knowledge and pest detection results from deep learning models, we have designed a set of sampling methods and population density calculation algorithms to solve the above-mentioned problems. First of all, we standardised the user's photo-taking process, that is, taking pictures at a distance of 30 cm from the target vertically. In order to achieve this, in the camera interface of our software, we use gyroscope to help users judge whether their shooting angle is vertical, and minimise the artificial error of the shooting distance through multi-point sampling. Then, we calculate the actual area of the photo by extracting the Exchangeable Image File (EXIF) information of the photo through the following equation:

$$S_{actual} = \frac{D_{target}}{F_{35mm}} \cdot 24 \times 36(mm^2), \tag{1}$$

where S_{actual} is the actual area of the single photos, D_{target} is the distance between the camera and the target. By balancing the clarity of the photo with the need to prevent insects from being disturbed by the close proximity, D_{target} was recommended as 30cm. However, D_{target} is not strict and can be adjusted by the user according to his/her own preferences, because benefiting from the threshold optimisation algorithm of Human-in-the-loop in chapter 3.2.5, the economic threshold of each user will be automatically fitted to his/her photographic habits. The larger the difference between the user's habits and the recommended D_{target}, the longer the fitting takes. $24 \times 36(mm^2)$ is the actual sensor area of a full frame camera. F_{35mm} is the "35mm equivalent focal length", which is the actual focal length of the current camera when converted to a full-frame camera. Because the sensor size of a full frame camera is fixed, and our sampling criteria fixes the distance between the object and the lens at 30cm, we only need the equivalent focal length of the current camera to calculate the actual area of the photo. Hence, we can calculate the population density of each species of pest in a single photo and in the entire field through the following equations:

$$\rho = \frac{n_{pest}}{S_{actual}} \tag{2}$$

$$\rho_{field} = \frac{1}{n_{photo}} \sum_{n_{photo}}^{i=1} \rho_i, \tag{3}$$

where ρ is the population density of a certain pest in a single photo, n_{pest} is the quantity of the pest, S_{actual} is the actual area of the photo, ρ_{field} is the population density of a certain pest in the entire field, and n_{photo} is the total number of samples taken in that field.

With the photos taken by the above photography standards, supplemented by the population density calculation algorithm we proposed, the system can link the data obtained from the sampling of mobile phone photography with the threshold value in agricultural prior knowledge for subsequent pest management decision making.

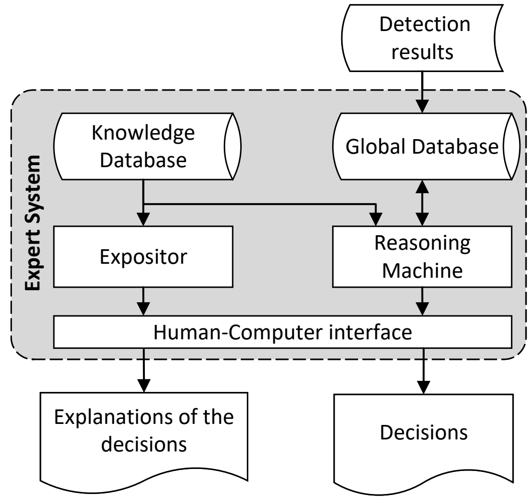


Figure 5: The proposed Integrated Pest Management decision making expert system

3.2.4. Rule-based reasoning expert system for pest management decision making

The calculation of the pest population density in the sampled photos provides a data basis for semi-automated IPM decision making. However, we still need to use relevant prior agricultural knowledge to conduct qualitative analysis on these data to make pest management decisions. There have been many studies (Dewar, Ferguson, Pell, Nicholls and Watts, 2016; Ellis, Berry, Walters et al., 2009; Wang, Bai, Zhao, Su, Liu, Han and Chen, 2020b; Wang, Zhao, Bai, Shang, Zhang, Hou, Chen and Han, 2021a; Gong, Li, Gao, Wang, Li, Zhang, Li, Liu and Zhu, 2021; Honek, Martinkova, Saska and Dixon, 2018) on the main invertebrate pests affecting wheat crops. However, the representation of such prior knowledge from the literature is usually text, which cannot be understood by computers. To address this problem, we developed an expert system that allows a prior knowledge of pests from the literature to be used to quantify the pest detection results obtained from the deep learning model.

Expert Systems are programme systems with expertise and experience that use the knowledge and experience provided by one or more experts in a particular field to reason and make judgements, simulate the decision making process of human experts, and use computers to automate the solution of complex problems that need to be handled by human experts. The rule-based expert system is currently the most commonly used method, mainly due to a large number of successful examples, as well as simple and flexible development tools. It directly imitates the human mental process and utilises a set of rules to represent expert knowledge.

In response to the above problems, we propose a rule-based expert system whose structure is shown in the figure 5. It consists of five parts: Global Database, Knowledge Database, Reasoning Machine, expositor and Human-Computer Interface. The Global Database is used to store initial data and intermediate data obtained during the decision making process. Specifically, it contains the species of insects in the pest detection results and their corresponding population densities. It also contains background information relevant to pest decision-making, such as time information extracted from the exif of insect photos, geographic coordinates, and weather information as well as crop type and growth stage information obtained through user input. The Knowledge Database stores the knowledge of domain experts in a certain storage structure, including facts and feasible operations and rules. Knowledge databases are constructed by computer experts in collaboration with domain experts. The computer experts represent the domain knowledge of the domain experts into a computer-understandable representation and store it in the knowledge database as rules. In this study, We summarised the thresholds about wheat pest management decision making in the previous literature (Dewar et al., 2016; Ellis et al., 2009; Wang et al., 2020b, 2021a; Gong et al., 2021; Honek et al., 2018) and normalised them into a computer-understandable Knowledge Database. It has an IF (condition) THEN (behaviour) structure. When the condition of the rule is met, the rule is triggered, and then make a decision. The Reasoning Machine selects matching rules from the Knowledge Database according to the input, and makes pest management decisions by executing the rules. The Expositor is used to explain the behaviour of the expert system to the user. The Human-Computer interface is used to display the decision results and their explanations.

Concretely, assuming that after sampling, detection, and population density calculation, the detection result indicates a population density of $100/m^2$ for grain aphids and $1/m^2$ for ladybirds. The software first extracts the time and geographic coordinates from the exif to determine the current weather, and then stores this information, along with user-entered information about crop type and growth stage, in the global database as the initial data for this decision. Then, the reasoning machine matches the initial data of this decision in the global database according to the rules in the knowledge database. The reasoning machine first determines the economic threshold of the pest by its species, crop species and crop growth cycle. Assuming that the growth cycle of the current crop wheat is in GS69: Flowering complete, the corresponding economic threshold of grain aphid in the knowledge base is $50/m^2$, and since the population density of grain aphid in the detection result meets the condition, the behaviour of the expert system is to recommend pesticide spraying at this time. Then, other contextual information is used to further adjust the decision. The first condition is the pest-beneficial insect ratio, the ratio of grain aphid to its natural enemy ladybird beetle is 10:1, which meets the corresponding condition in the knowledge database, so the recommendation of pesticide spraying is kept unchanged: and then it is the weather condition, assuming that it is a rainy day, the operation will change the recommendation to delay pesticide spraying. At the same time, Expositor summarises the decision-making process and presents it to the user via HCI output.

3.2.5. Human-in-the-loop threshold optimisation algorithm

Although we have obtained some thresholds from the literature, the above work is still not sufficient for a pest management decision making system. There are a number of reasons for this: First of all, not all crops have known thresholds for each pest in each growth stage. For example, there is no known threshold for gout fly in spring cereals, despite the high risk of yield reduction (Ellis et al., 2009; Dewar et al., 2016). Second, because some studies were conducted a long time ago (more than ten years ago), their pest thresholds may not still be applicable today. last but not least, pests will lead to increased resistance to pesticides after natural selection, so we cannot use a constant threshold for pest management in the future.

To keep the thresholds up-to-date in our pest management expert system, we designed a human-in-the-loop threshold optimisation algorithm. Human-in-the-loop (HITL) is a branch of artificial intelligence in which people participate in a virtuous circle in which they train, adapt and test specific algorithms to improve the accuracy of the model.

As shown in figure 6, each time a user makes a pest management decision using the software, the server, in addition to recording the decision, sends the user a questionnaire after a certain interval (the length of this interval varies from a few hours to a few days, depending on how quickly the operation used takes effect) asking the user to observe the farm to determine the effectiveness of the last decision. The system then automatically adjusts certain thresholds in the database based on the effectiveness of the last decision.

For instance, continuing with the example from chapter 3.2.4, assuming that the expert system gives a decision

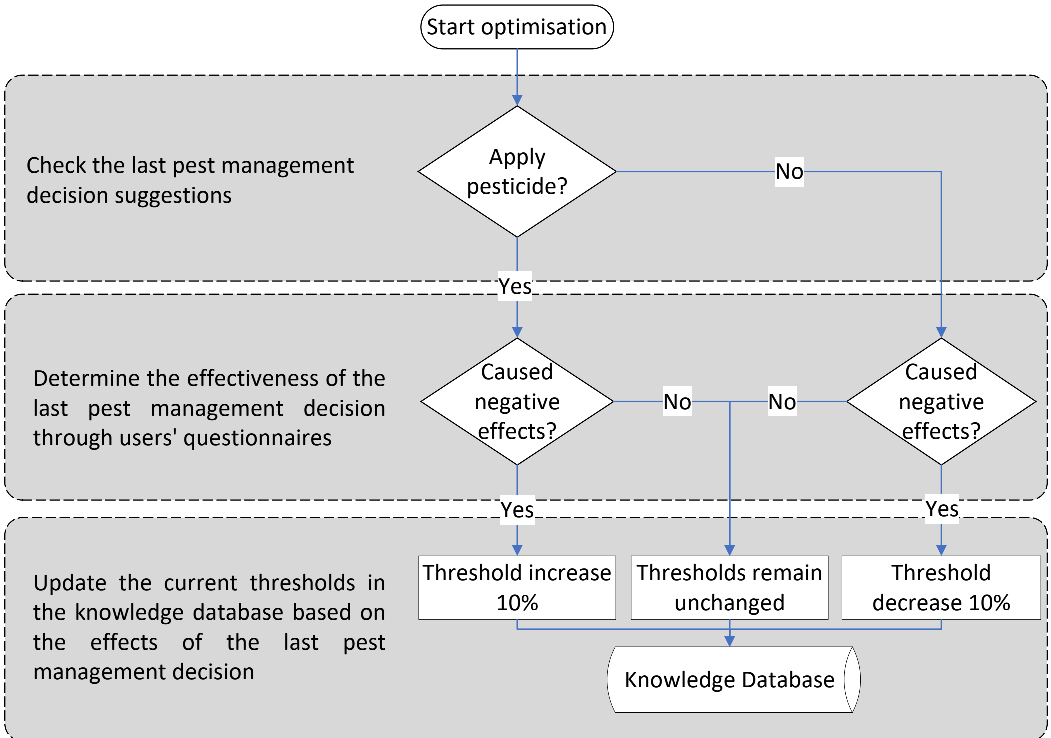


Figure 6: The pipeline of the proposed HITL thresholds optimisation algorithm

suggestion to use cypermethrin for control, the server will push a questionnaire to the user. The questionnaire will ask the user to take another sampling in the field to calculate the ratio of natural enemies to grain aphids as a criterion, if the ratio decreases, it means that the decision has a negative effect, this is because the existing economic threshold in the expert system is set too low, so the threshold will be automatically optimised to the existing threshold by increasing it by 10%, on the contrary, it means that the decision does not have a negative effect and the threshold will remain unchanged. If the advice given by the expert system is not to apply pesticides, the server will send a questionnaire to the user after one week to go to the field and observe if the phenomenon of yellow dwarf disease occurs in wheat. If there is a negative impact such as yellow dwarf disease, the threshold in the expert system is too low, resulting in the threshold not being triggered in time, and the system will automatically lower the existing threshold by 10%. On the contrary, if there is no negative effect, the existing threshold will be lowered by 10%.

4. Results and discussions

4.1. Evaluation Metrics

Multiple metrics are used to evaluate the object detection model, including mean average precision (mAP), the number of frames dealt within a second (FPS), and the number of parameters (Parameters) in the detection model. The mean average precision is a general evaluation metric for object detection model, which is defined as the mean value of the area under the Precision-Recall (PR) curve ,

$$Pr(n) = \frac{TP_n}{TP_n+FP_n} \tag{4}$$

$$Re(n) = \frac{TP_n}{TP_n+FN_n} \tag{5}$$

$$mAP = \frac{1}{N}\sum_{n\in N}\int_0^1 Pr(n)d\,Re(n) \ , \tag{6}$$

where N is the number of object categories, TP_n, FP_n, and FN_n refer to the number of true positive samples, false positive samples, and false negative samples for class n, respectively. The true positive samples in object detection tasks are defined by intersection over union (IoU), which is a ratio of the overlap area in the union area between the predicted bounding box and the annotated bounding box. Parameters metrics measure the size of the object detection model. The larger object detection model requires more computational resources.

4.2. Performance Evaluation of the Detection Model

We evaluate the performance of the detection model using three pest datasets, including IP102, AgriPest, and Our Dataset. Each dataset is divided into a training dataset, validation dataset, and test dataset in a ratio of 8:1:1. The mAP for each trained model on the test dataset is calculated and is presented in Table 2. The compared models are pre-trained on the COCO dataset (Lin, Maire, Belongie, Hays,

Figure 7: Qualitative results of the pest detection model on our dataset. The detection results demonstrate the ability of our model to accurately identify multiple tiny pests in one image.

Perona, Ramanan, Dollár and Zitnick, 2014). As mentioned before, multiple data augmentation methods are used in the training dataset. The dropout method are used to avoid overfitting.

Table 2
The performance for different detection models

	Faster RCNN	YoloX	Our Model
FPS	11.45	12.97	13.21
Parameters	28,275k	8,976k	6,759k
mAP (IP102)	55.25%	56.87%	58.17%
mAP (AgriPest)	7.18%	66.24%	75.29%
mAP (Our dataset)	24.21%	54.26%	57.33%

As Table 2 shown, we compare our model with Faster RCNN and YoloX on the multiple pest dataset. The mAP of different models is mainly limited by the challenge of the pest detection task. Although the object detection models do not present surprising performance in terms of mAP, our model outperforms Faster RCNN and YoloX due to it adopting the Path Aggregation Network to fuse multi-scale features. In particular, our model obtains mAP of 75.29% and 57.33% on the AgriPest and our datasets, respectively. The failure of Faster RCNN is due to the challenge of tiny objects in AgriPest and our datasets. Meanwhile, YoloX and our model achieve faster detection speed with fewer training parameters than Faster RCNN. The main difference between our model and YoloX is a more efficient neck and data augmentation methods for pest detection. Figure 7 presents the detection results of our model. In summary, our model achieves state-of-the-art results in pest detection tasks.

4.3. Qualitative In-Field Validation of the System Usability

In order to validate the usability of the proposed method, an in-field testing was conducted. Figure 8 illustrates the flow of our in-field experiments. Following the selection of the field to be tested, the tester took images at the sampling points generated by the ERCS algorithm. Upon completion of each picture collection, the mobile application output the detection results of the pest detection model to the user. Once all sampling points had been evaluated, the expert system determined the pest severity and provided pest management advice. The application was deployed on a range of mobile phones, equipped with mid-end (Qualcomm Snapdragon 855, 875, 8Gen1, etc.) or low-end (Qualcomm Snapdragon 695, 720G, etc.) system-on-a-chips (SoCs), in order to assess its usability across a spectrum of computing power platforms. A series of experiments was conducted at multiple sites in England. The experimental sites were located in West Yorkshire (Leeds and Knottingley), North Yorkshire (Malton), Warwickshire (Nuneaton), and Nottinghamshire (Mansfield). A total of 12 testers participated in the experiments. The qualitative validation of the usability of the proposed methods with the developed mobile application on mobile phones with different performances and in various regions of England was achieved.

4.4. Quantitative Evaluation of the User Usability

The usability of the proposed solution rely on the friendliness of user interface and function design, in addition to the stability of the system. The mobile application provides end users with the ability to browse farm information, add farm records, respond to tasks, detect pests, view weather forecast, modify app settings and more. Meanwhile, a manually collected encyclopaedia of knowledge about pests and crops is integrated as a knowledge base for providing the basic knowledge and advice for model decisions in the integrated pest management function. The above functional design is based on a user requirements analysis of the system in early stage. The usability evaluation process invites end users to make subjective evaluations of the functionality of the mobile application, the efficiency and accuracy of the functions, and the user-friendliness of the interface. Specifically, evaluation participants were asked to follow an instructional document after logging into the app to complete their experience of the functions in the mobile application and to rate the usability of key functions. The results are shown in Table 3.

In addition to the evaluation of functional usability, an open access experiment which invite participants to use the application without restrictions was processed. The results of this experiment was collected by the System Usability Scale (SUS) questionnaire, which consists of ten questions with a scale from strongly agree (5 points) to strongly disagree (1 point) for each question (Lewis, 2018). The

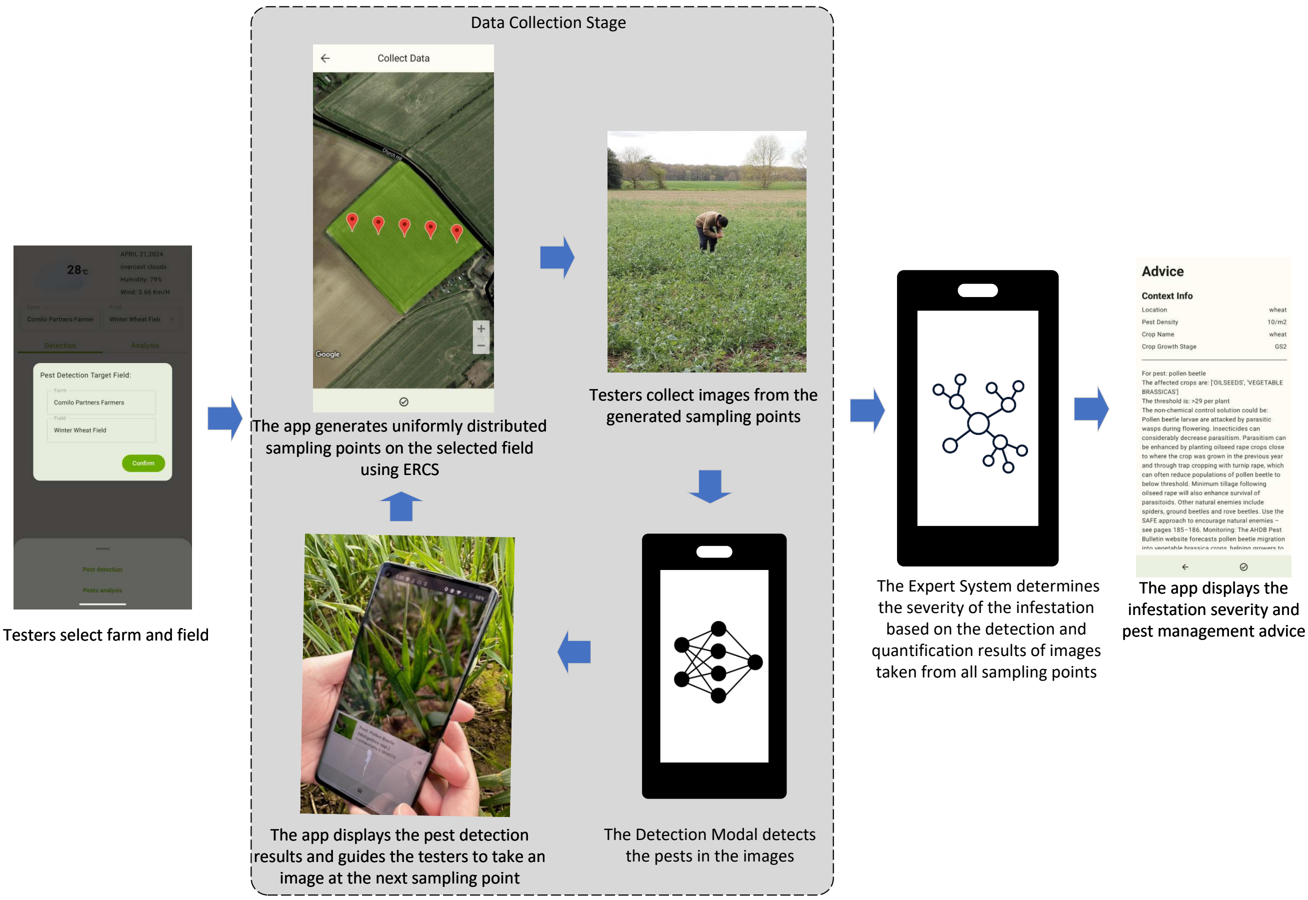


Figure 8: Flowchart of the in-field experiment to test the mobile application.

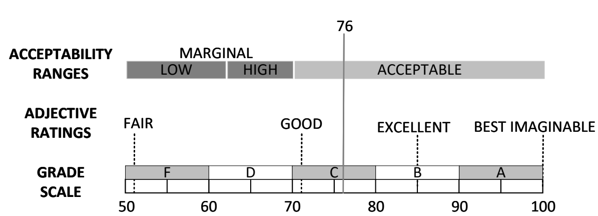


Figure 9: SUS Score of the our mobile application

questions in the SUS questionnaire focus on the system usability, such as, I think that I would like to use this system frequently, and I needed to learn a lot of things before I could get going with this system. The final evaluation results are calculated according to Equation 7, where $S1$ to $S10$ indicate the scoring of each of the 10 questions.

$$\begin{aligned}\text{SUS} = 2.5 \times (20 + \text{SUM}(S1, S3, S5, S7, S9) - \\ \text{SUM}(S2, S4, S6, S8, S10))\end{aligned} \tag{7}$$

Figure 9 presents the results for the open access experiment. According to the grading based on SUS scores (Lewis, 2018), the mobile application with an average score of 76 is considered as a good product.

Table 3
Questionnaire results for User Experience Tasks (Including login, fields information, record, task, detection, maps, weather, the encyclopaedia and IPM)

Task	1 (Hard)	2	3	4	5 (Easy)
Task 1	0%	0%	9%	16%	75%
Task 2	0%	0%	14%	18%	68%
Task 3	0%	0%	10%	7%	83%
Task 4	9%	0%	9%	9%	73%
Task 5	8%	3%	16%	9%	64%
Task 6	7%	8%	6%	12%	67%
Task 7	9%	5%	4%	9%	73%
Task 8	0%	0%	0%	18%	82%
Task 9	0%	0%	18%	0%	82%

5. Conclusion and Future Work

In this work, we develop a practical application of an end-to-end decision making system for integrated pest management that allows users to take just a few photos to get pest management advice, enabling growers with no agricultural knowledge to apply sustainable crop protection. The present study has offered a framework which integrated deep learning objective detection and expert system for the exploration of environmentally friendly pest management thresholds for wheat. In this study, we proposed a low computational cost sampling point generation algorithm that enables mobile devices to generate evenly distributed sampling points in arbitrary-shaped farmlands. We used PyTorch Mobile to generate a lightweight pest detection model that can be deployed on mobile devices, so that our application can get rid of the constraints of communication infrastructure. We have developed a standardised sampling protocol and used our software to assist users with sampling, enabling the calculation of pest population densities from a single photograph. A rule-based expert system has been established for deriving pest management thresholds from prior agriculture knowledge and making decisions based on pest detection results. We proposed a human-in-the-loop algorithm to continuously track the validity of thresholds in the expert system and keep them up-to-date.

The experimental results show that our detection model outperformed Faster RCNN and YoloX in term of FPS and mAP. In the user evaluation of system usability, the proposed system received 76 in SUS score. An in-field qualitative evaluation of system usability has also been conducted.

A number of limitations need to be noted regarding the present study:

Firstly, our current population density calculation is achieved by hard-coding the distance from the lens to the target, which is the result of the compromise of many factors, although the computational cost is lower and the generality is better, but it also leads to a greater error in the calculation of population density. Therefore, we intend to develop a low-computational cost AI distance measurement algorithm to replace the existing hard-coding method to improve the accuracy of population density calculation in our subsequent research.

Secondly, the decision-making expert system has only been validated for usability, while the validation of its determination of the severity of pest infestation and the feasibility of the generated pest management advice still requires further research. In future work, it would be beneficial to conduct interdisciplinary research with agronomists conducting pest threshold studies and entomologists conducting pesticide resistance studies.

Thirdly, our human-in-the-loop threshold optimisation algorithm have not been validated for the time being as this would take many years of experimentation over multiple crop cycles to complete. In terms of this direction for future research, the validation of the threshold optimisation algorithm in practice is required to confirm the effectiveness of our proposed solution.

References

Agriculture and Horticulture Development Board, 2022. Planting and variety survey. URL: https://ahdb.org.uk/planting-variety-survey.

Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 .

Curtis, B.C., Rajaram, S., Gómez Macpherson, H., et al., 2002. Bread wheat: improvement and production. Food and Agriculture Organization of the United Nations (FAO).

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection, in: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), Ieee. pp. 886–893.

Dedryver, C.A., Ralec, A.L., Fabre, F., 2010. The conflicting relationships between aphids and men: A review of aphid damage and control strategies. Comptes Rendus Biologies 333, 539–553. URL: https://www.sciencedirect.com/science/article/pii/S1631069110001150, doi:https://doi.org/10.1016/j.crvi.2010.03.009.

Department for Environment, Food & Rural Affairs, 2020. Farming statistics - final crop areas, yields, livestock populations and agricultural workforce at 1 june 2020- uk. URL: https://www.gov.uk/government/statistics.

Dewar, A.M., Ferguson, A., Pell, J.K., Nicholls, C., Watts, J., 2016. A review of pest management in cereals and oilseed rape in the uk. Research Review .

El-Azhary, E.S., Hassan, H.A., Rafea, A., 2000. Pest control expert system for tomato (pcest). Knowledge and Information Systems 2, 242–257.

Elfwing, S., Uchibe, E., Doya, K., 2018. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural Networks 107, 3–11.

Ellis, S., Berry, P., Walters, K., et al., 2009. A review of invertebrate pest thresholds. HGCA Research Review .

Ellis, S., White, S., Holland, J., Smith, B., Collier, R., Jukes, A., 2014. Encyclopaedia of pests and natural enemies in field crops .

Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J., 2021. Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430 .

Gerevini, A., Perini, A., Ricci, F., Forti, D., Ioriatti, C., Mattedi, L., Monetti, A., et al., 1992. Pomi: an expert system for integrated pest management of apple orchards .

Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 580–587.

Gong, P., Li, X., Gao, H., Wang, C., Li, M., Zhang, Y., Li, X., Liu, E., Zhu, X., 2021. Field evolved resistance to pyrethroids, neonicotinoids, organophosphates and macrolides in rhopalosiphum padi (linnaeus) and sitobion avenae (fabricius) from china. Chemosphere 269, 128747.

Harris, C., Stephens, M., et al., 1988. A combined corner and edge detector, in: Alvey vision conference, Citeseer. pp. 10–5244.

Harrison, S.R., 1991. Validation of agricultural expert systems. Agricultural systems 35, 265–285.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Honek, A., Martinkova, Z., Saska, P., Dixon, A.F., 2018. Aphids (homoptera: Aphididae) on winter wheat: predicting maximum abundance of metopolophium dirhodum. Journal of Economic Entomology 111, 1751–1759.

Kaloudis, S., Anastopoulos, D., Yialouris, C.P., Lorentzos, N.A., Sideridis, A.B., 2005. Insect identification expert system for forest protection. Expert Systems with Applications 28, 445–452.

Lewis, J.R., 2018. The system usability scale: past, present, and future. International Journal of Human–Computer Interaction 34, 577–590.

Leybourne, D.J., Storer, K.E., Berry, P., Ellis, S., 2022. Development of a pest threshold decision support system for minimising damage to winter wheat from wheat bulb fly, delia coarctata. Annals of Applied Biology 180, 118–131.

Liao, S.H., 2005. Expert system methodologies and applications—a decade review from 1995 to 2004. Expert systems with applications 28, 93–103.

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context, in: European conference on computer vision, Springer. pp. 740–755.

Liu, S., Qi, L., Qin, H., Shi, J., Jia, J., 2018. Path aggregation network for instance segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8759–8768.

Mottaleb, K.A., Kruseman, G., Snapp, S., 2022. Potential impacts of ukraine-russia armed conflict on global wheat food security: A quantitative exploration. Global Food Security 35, 100659.

Nancarrow, N., Aftab, M., Hollaway, G., Rodoni, B., Trębicki, P., 2021. Yield losses caused by barley yellow dwarf virus-pav infection in wheat and barley: A three-year field study in south-eastern australia. Microorganisms 9, 645.

Rafea, A.A., El-Azhari, S., Ibrahim, I., Edres, S., Mahmoud, M., Street, E.N., 1995. Experience with the development and deployment of expert systems in agriculture., in: IAAI, pp. 146–155.

Redman, G., 2022. John Nix Pocketbook for farm management 53rd edition. AGRO BUSINESS CONSULTANTS.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28.

Sarkozi, A., 2019. New standards to curb the global spread of plant pests and diseases. URL: https://www.fao.org/news/story/en/item/1187738/icode/.

Senevirathna, K.M., Guelly, K.N., Mori, B.A., 2023. Management of the orange blossom wheat midge, sitodiplosis mosellana, in western canada. Plant Health Cases , phcs20230002.

Shewry, P.R., 2009. Wheat. Journal of experimental botany 60, 1537–1553.

Shimrat, M., 1962. Algorithm 112: Position of point relative to polygon. Commun. ACM 5, 434. URL: https://doi.org/10.1145/368637.368653, doi:10.1145/368637.368653.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 .

Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W., 2013. Selective search for object recognition. International journal of computer vision 104, 154–171.

Wang, C.Y., Liao, H.Y.M., Wu, Y.H., Chen, P.Y., Hsieh, J.W., Yeh, I.H., 2020a. Cspnet: A new backbone that can enhance learning capability of cnn, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pp. 390–391.

Wang, K., Bai, J., Zhao, J., Su, S., Liu, L., Han, Z., Chen, M., 2020b. Super-kdr mutation m918l and multiple cytochrome p450s associated with the resistance of rhopalosiphum padi to pyrethroid. Pest Management Science 76, 2809–2817.

Wang, K., Zhao, J.N., Bai, J.Y., Shang, Y.Z., Zhang, S.Q., Hou, Y.F., Chen, M.H., Han, Z.J., 2021a. Pyrethroid resistance and fitness cost conferred by the super-kdr mutation m918l in rhopalosiphum padi (hemiptera: Aphididae). Journal of Economic Entomology 114, 1789–1795.

Wang, R., Liu, L., Xie, C., Yang, P., Li, R., Zhou, M., 2021b. Agripest: a large-scale domain-specific benchmark dataset for practical agricultural pest detection in the wild. Sensors 21, 1601.

Wu, X., Zhan, C., Lai, Y.K., Cheng, M.M., Yang, J., 2019. Ip102: A large-scale benchmark dataset for insect pest recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8787–8796.

Yialouris, C., Sideridis, A., 1996. An expert system for tomato diseases. Computers and electronics in agriculture 14, 61–76.

Yuan, Z., Li, S., Yang, P., Li, Y., 2022. Lightweight object detection model with data augmentation for tiny pest detection, in: 2022 IEEE 20th International Conference on Industrial Informatics (INDIN), IEEE. pp. 233–238.