Neural Computing and Applications (2024) 36:20211–20229
https://doi.org/10.1007/s00521-024-10142-0

ORIGINAL ARTICLE

Advancing precision agriculture: domain-specific augmentations and robustness testing for convolutional neural networks in precision spraying evaluation

Harry Rogers[1] · Beatriz De La Iglesia[1] · Tahmina Zebin[2] · Grzegorz Cielniak[3] · Ben Magri[4]

Received: 7 October 2023 / Accepted: 27 June 2024 / Published online: 12 August 2024

Abstract
Modern agriculture relies heavily on the precise application of chemicals such as fertilisers, herbicides, and pesticides, which directly affect both crop yield and environmental footprint. Therefore, it is crucial to assess the accuracy of precision sprayers regarding the spatial location of spray deposits. However, there is currently no fully automated evaluation method for this. In this study, we collected a novel dataset from a precision spot spraying system to enable us to classify and detect spray deposits on target weeds and non-target crops. We employed multiple deep convolutional backbones for this task; subsequently, we have proposed a robustness testing methodology for evaluation purposes. We experimented with two novel data augmentation techniques: subtraction and thresholding which enhanced the classification accuracy and robustness of the developed models. On average, across nine different tests and four distinct convolutional neural networks, subtraction improves robustness by 50.83%, and thresholding increases by 42.26% from a baseline. Additionally, we have presented the results from a novel weakly supervised object detection task using our dataset, establishing a baseline Intersection over Union score of 42.78%. Our proposed pipeline includes an explainable artificial intelligence stage and provides insights not only into the spatial location of the spray deposits but also into the specific filtering methods within that spatial location utilised for classification.

Keywords Agri-Robotics · Computer vision · Data augmentation · XAI

1 Introduction

Agricultural sprayers are essential tools for crop protection and management, as they enable farmers to distribute fertilisers, herbicides, and pesticides over large areas of land. However, the efficient and effective use of these sprayers relies heavily on an accurate assessment of their performance, particularly in terms of identifying the locations where spray deposits have landed. Traditionally, this has been done manually, with methods such as tracers and water-sensitive papers (WSPs). Both methods require manual input from farmers and only approximate spray deposit deposition values on targets or non-targets. Recent work by [1] shows that system evaluation is required but

Harry Rogers and Beatriz De La Iglesia have contributed equally to this work.

✉ Harry Rogers
Harry.Rogers@uea.ac.uk

✉ Beatriz De La Iglesia
b.iglesia@uea.ac.uk

✉ Tahmina Zebin
tahmina.zebin@brunel.ac.uk

✉ Grzegorz Cielniak
gcielniak@lincoln.ac.uk

✉ Ben Magri
ben.magri@syngenta.com

[1] School of Computing Science, University of East Anglia, Norwich Research Park, Norwich, Norfolk NR4 7TJ, UK

[2] School of Computer Science, Brunel University London, Kingston Ln, London, Uxbridge UB8 3PH, UK

[3] School of Computer Science, University of Lincoln, Brayford Pool, Lincoln, Lincolnshire LN6 7TS, UK

[4] Research, Syngenta ltd, Jealott's Hill, Bracknell, Warfield RG42 6EY, UK

tracers are used for such evaluation, and they have shortcomings.

In this work, we propose a methodology that can detect spray deposits using camera images from precision agricultural sprayers without relying on tracers or WSPs. Because there is a lack of comprehensive solutions for this challenge, we conducted our own data collection using a precision spot spraying system equipped with a high-quality camera and trays of lettuce and target weeds. Our prior proof-of-concept work [2, 3] introduced a number of approaches to classify spray deposits without tracers or water-sensitive papers. To address some of the limitations in classification accuracy and robustness of the approaches we developed, we devised and tested a number of novel augmentation methods to augment our initially collected dataset. Additionally, we introduce a weakly supervised object detection (WSOD) process specific to Agri-Robotic settings.

We employ an explainable artificial intelligence (XAI) pipeline to assess convolutional neural networks (CNNs) for the purpose of classifying and detecting spray deposits on target weeds and non-target lettuce. Using our dataset, we started with a binary classification task, distinguishing between sprayed and dry target weeds and non-target lettuce. We used multiple pre-trained convolutional backbones such as the EfficientNetB0, DenseNet121, ResNet18, and VGG11 as our feature extractor before training our domain-specific classification head. Once classified, effective models generate class activation maps (CAMs), which are heatmaps illustrating a CNN's focus areas within the image. These CAMs are generated by visualising the focal regions from the final convolutional layer. We assess these CAMs using XAI metrics such as deletion, insertion, and a uniquely proposed weakly supervised object detection (WSOD) process. These metrics help determine if a CAM effectively explains the features a CNN utilises for inference. As we also employed novel augmentation methods, we use classification metrics and a novel robustness testing methodology using generated CAMs as a starting point. This process allows for further insights into what a CNN is using as an important filtering method and provides us with evaluation metrics to gauge the effectiveness of the applied augmentations in assessing the robustness of the system we devised.

In the review of the related literature for this work, we identify a research gap in automating the assessment of precision sprayers. We also observe the use of stratified sampling methods used by others to enhance neural network (NN) learning, to explain our stratified sampling methodology. Furthermore, we examine data augmentations, starting with general deep learning techniques applicable across various tasks and domains, and then looking more into domain and task-specific methods. Our review suggests that incorporating novel, domain-specific augmentation methods can significantly enhance the overall effectiveness of fully developed vision systems.

From the XAI pipeline and robustness testing methodology developed during this work, we find that our novel augmentations improve classification scores and vastly improve all robustness augmentation scores. From the robustness testing stage, the top three filtering methods that have the highest likelihood used within spatial locations for CNN inference are identified.

Our contributions are as follows:

- We have proposed a number of novel data augmentation methods specific to agricultural field settings. We obtained improved classification scores and robustness with domain-specific augmentation when compared to a baseline model.
- We compiled an XAI pipeline with a novel robustness testing approach, which assesses the proposed data augmentation methods and demonstrates the effectiveness of our custom solution.
- Through this robustness testing approach, the developed pipeline can pinpoint potential features within the spatial regions of the image that CNNs utilise for classification, enhancing the comprehension of the inference process of the trained CNN.

The subsequent sections of this paper are structured as follows. In Sect. 2, we review pertinent literature concerning precision spraying evaluations, stratified sampling methodologies, and data augmentations. Section 3 provides comprehensive insights into our XAI pipeline, the feature-based stratified sampling method, and additional implementation details. In Sect. 4, we introduce the dataset and the novel augmentation methods designed explicitly for this domain-specific dataset. The evaluation and reporting of the developed CNNs' performance, CAM explanations, and the robustness of our novel augmentation methods are discussed in Sect. 5. Lastly, in Sect. 6, we summarise the paper's conclusions and outline potential avenues for future research.

2 Related work

To demonstrate the need of an automated assessment for precision spraying, we review current evaluation methodologies. Currently within Agri-Robotics, existing literature is saturated primarily with the detection of weeds and crops using CNNs with success [4–6]. However, existing systems lack information on the evaluation of spraying accuracy without the use of traditional agricultural methods.

Following this, we identify stratified sampling and data augmentations as instrumental techniques used to improve the effectiveness of machine and deep learning algorithms. These methods help in mitigating issues related to data availability. However, existing approaches for data augmentation and stratified sampling may not always provide satisfactory results. Consequently, in our review, we identify stratified sampling methodologies from multiple domains for a variety of tasks. Moreover, in our review of data augmentations, we look at general use, domain, and task-specific methods and show a need for domain-specific methods for Agri-Robotics. From our review of data augmentations, robustness testing can also be identified.

2.1 Precision spraying evaluation

Within precision spraying, three main methodologies have been used to evaluate the effectiveness of spraying within fields. Proposed precision sprayers use one of these to evaluate the spraying of the proposed systems.

The most common method is WSPs. WSPs are yellow pieces of paper that change colour when they are sprayed [7], an example is shown in Fig. 1. Many types of ground and aerial spraying systems have been developed to complete precision spraying and evaluated with WSPs. Example applications include orchard tree spraying [8–10], disease detection in potatoes [11], weed spraying in corn fields, cabbage fields, and cereal fields [12–14]. General testing for aerial-based spraying has been primarily explored with the usage of WSPs [15, 16]. However, there are several drawbacks with WSPs. Firstly, a person has to put the WSP out to be sprayed and then retrieve it for analysis after spraying. Secondly, the texture of WSPs is not the same as the targets and, therefore, will act differently. Finally, no spraying system when deployed can perfectly recreate the same spray deposit; therefore, the spray on the WSP will be different from what is sprayed on a target or non-target.

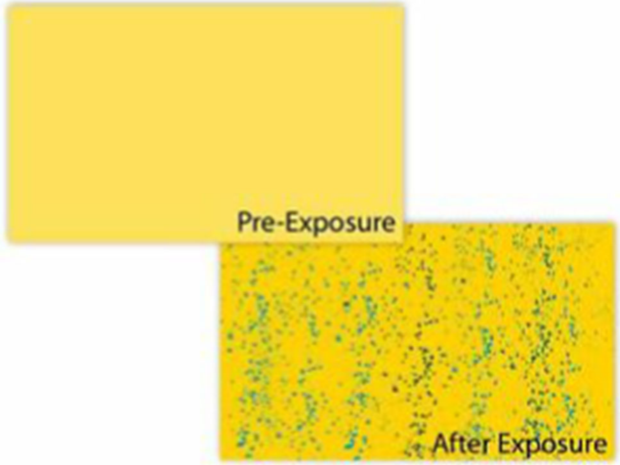


Fig. 1 Example of WSPs

The usage of WSPs can be improved upon with tracers, tracers are dyes that are added to the spraying chemical to make it abundantly clear where deposits have landed. This means that the actual spray deposits from each system proposed can be evaluated. Much like WSPs tracers have been explored within a range of applications such as pesticide spraying in maize fields, orchards, and rice fields [17–19]. Weed control has also been explored in field and in controlled testing environments [20–22]. However, systems do not use the same tracer as not all are available for use when mixed with differing types of chemical applications. Tracers may also need to be harvested in some cases to be analysed which may be too late or may require specialist equipment to analyse.

Finally, some systems are evaluated with human intervention where humans will count the number of weeds sprayed. These systems typically use an AI detection system to identify weeds and spray them [23, 24]. Some other systems calculate the volume of chemical sprayed [25, 26]. However, the intervention of humans creates ambiguity where weeds can be partially sprayed. This creates more problems as this does not describe an accurate system. When considering volume calculations, these can be skewed, and many commercial systems sell on this factor.

From this review, it can be stated that precision spraying systems need an automated evaluation system that is effective and robust. There needs to be a system that can evaluate without the usage of WSPs or tracers. Therefore, we have developed an XAI system that can do so with the usage of stratified sampling and data augmentations to increase robustness and accuracy.

2.2 Stratified sampling

To allow machine and deep learning algorithms to excel in terms of accuracy, sampling and stratification methodologies have been employed due to domain-specific issues regarding useful data. Stratification refers to organising data into specific groups based on a relevant criterion. Whilst sampling takes specific data points and uses them for training algorithms effectively. This can be through showing a diverse set of training points for an algorithm to learn from.

Within the Agri-Robotics, domain sampling methods have been developed to aid CNNs in their learning within the object detection task. For example, the system from [27] blends region convolutional neural networks and feature pyramid networks to select images from a robotic system. Once images are collected, t-distributed stochastic neighbour embedding is applied to select images with similar features. After clustering with K-means, similar images are used to aid NN learning. The proposed method clusters crops that are of the same class without the need for human intervention. The stratified sampling method is compared to several sampling methods and drastically improves from these baselines. With this method of classification and object detection, images can be paired without the need for human intervention which then can be used for effective NN learning. However, clustering our data would cause images that are sprayed and dry to be clustered together as the images are nearly identical, and the only changes are from crops and weeds being in different spatial locations. This may not be useful to our specific task of spray detection within the images.

In the medical imaging domain, sampling methods have been developed with selective sampling. The selective sampling method proposed by [28] aids in haemorrhage detection by dynamically selecting misclassified negative samples during training. Training samples are heuristically selected based on classification confidence by the current status of the CNN. Weights are assigned to the training samples so that the most informative samples are likely to be included in the next CNN training iteration. This method is very effective at identifying small differences in a binary classification scenario. However, as our images have a high resolution to be able to identify spray deposits, this would be computationally expensive. Thus, we have proposed a simplistic methodology to pair images together for CNN learning.

Stratified sampling can also be used to create memory-efficient systems in more complex environments such as a 3D environment [29] and videos [30]. Within the 3D environment, stratified sampling is primarily used due to the network architecture, to make the overall system memory efficient. Within the proposed methodology from [29], the stratification identifies distant points in 3D space as keys with a sparse encoding. The proposal serves to expand the model's effective receptive field and establish direct long-range dependencies, whilst incurring minimal computation. The method is compared to other components used and is the most successful when the stratified sampling methodology is used. On the other hand, [30] uses their sampling methodology to find the most useful frames from videos to complete the classification of human actions. [30] use a Gaussian weighing function to aggregate the frames into a smaller subset which results in excellent classification results. Despite both methods from [29, 30] being effective in their respective scenarios, in our scenario, these both would not work. Reducing the dataset from a Gaussian weighing function is only useful when there is too much data that could be considered noisy, and usage of a 3D methodology is difficult when the data we have are currently 2D.

Traditionally, stratified sampling is employed to address class imbalances within data and cross-validate results, thereby increasing the effectiveness of developed systems [31]. Instead, in our approach, we will utilise stratified sampling to train CNNs in recognising image-specific features within a binary dataset. From the literature, it has been observed that creating a stratified sampling methodology with minimal computational overhead proves effective in aiding classification and NN learning. Various reviewed papers demonstrate improvements through their respective sampling methodologies. To validate the effectiveness of our proposed method, feature-based stratified sampling, we will adopt a modular approach and compare it to random stratified sampling, and random sampling without stratification.

2.3 Data augmentation

Data augmentation is a useful methodology for helping fix class imbalances, improving NN learning, robustness, and creating more accurate systems. There are many types of image augmentation that can be applied, we will explore some of the most popular and useful augmentations.

Geometric transformations such as translation, rotation, flip, and cropping are used throughout a wide variety of datasets that are publicly available [32]. These augmentations are not task-specific or domain-specific, yet they can be applied to not only classification but also other complex tasks like object detection; these are general augmentations. General augmentations are used to add more variation in datasets for NN learning when data are limited and hard to acquire. This applies to situations where data entries may be rare or costly to locate, such as in the context of medical data, where high-risk patient classes are often under-represented due to their lower prevalence.

Image augmentations can be domain-specific, for example, within the medical domain, image augmentations are imperative to ensure NNs can identify areas that can be used for the correct classification and detection of diseases [33]. NNs cannot learn well enough with limited datasets so augmentation becomes necessary to increase data availability, otherwise, within the medical domain, for example, poor models can lead to misdiagnosis. In Agri-Robotics, there have been some non-domain-specific augmentations to data [34]. [34] consider some non-domain-specific augmentations for agriculture with colour space, generative adversarial networks (GANs), and geometric techniques. In our work, using a GAN will likely result in more dry images as the spray deposits are hard to replicate as they are non-uniform in shape and size. Changing the colour space does not seem worthwhile as the targets will be a specific colour when deployed and will need to be located. Finally, if we could change the shape of spray deposits on target weeds with an automated method that would require a system to identify the deposits first.

[35] use data augmentations to aid learning for surface defect detection that could be considered similar to our problem of finding spray deposits on leaf surfaces. Surfaces for defect detection have been augmented using GANs to help improve scores across multiple metrics for CNNs. In the work by [35], the surfaces are supposed to be uniform in shape with a consistent texture. However, our problem has very small spray deposits, which are hardly visible with non-uniform shapes and on leaves with non-uniform textures so the approach may not be adequate.

We utilise slicing-aided hyper inference (SAHI) [36] to augment our data. Currently, SAHI is used to slice an image into a set number of sub-images that are then used for either inference or training. Pre-trained and fine-tuned networks improve when compared to their respective baselines when using SAHI. This is used for the detection of small objects within images which is a similar scenario to this work as spray deposits on leaves are very small. However, this augmentation has not been used for just image classification. Our usage is also to augment the dataset to aim to create closer to realistic row-wise images of crops and weeds with one crop per image.

When considering the robustness of CNNs, colour transformations can be considered to improve robustness to lighting changes. In the context of adversarial networks, robustness experiments have been conducted to identify robust methods for data augmentation [37]. Such methods apply augmentations to an image in an attempt to trick a network into an incorrect prediction. In our work, we use unique ways of robustness testing to evaluate the addition of novel augmentations. Robustness can, in our context, also be used to identify features that are more likely used in CNN prediction.

In this paper, we discuss the significance of lightweight data augmentations that not only enable efficient computation but also improve NN learning. To accomplish this, we introduce an innovative approach that harnesses domain-specific information by using images taken both before and after spraying. By subtracting these two images, we can pinpoint areas of interest where spray deposits are situated. Furthermore, we acknowledge the importance of concentrating on areas containing green leaves, as the crops and weeds in our dataset share this characteristic. Consequently, we explore and employ both of these methods, we presented our findings on this in greater detail in Sect. 4.

3 XAI pipeline

The proposed XAI pipeline, as shown in Fig. 2, consists of multiple modular components. We begin with data collection and preprocessing. Next, we test our feature-based stratified sampling method against random sampling, stratified sampling, and random sampling with a random data split. After modifying pre-trained CNNs to create a binary classification head, we assess the sampling methods using classification metrics to determine the most effective one. Once this step is completed, we introduce new augmentations and train CNNs using the best sampling method, we interpret our CNN's decisions through Grad-CAM++ [38] visualisations and evaluate these with deletion, insertion, and WSOD [39]. Finally, we assess the robustness of our augmentations using our proposed methodology. Using this pipeline means that there are no parameters to tune or learn as all methods, apart from training the model, are statistical methodologies used to evaluate the trained model. Following the original data split, we will also evaluate with a tenfold cross-validation, ensuring the original data split is not one of the folds, and use all metrics in the evaluation apart from WSOD.

3.1 Feature-based stratified sampling

Stratification is the process of dividing data into subgroups based on a criterion. For our dataset, we split our quadrant images into two strata using the binary classification labels sprayed or dry.

Let $x_1, x_2, .., x_n$ be our quadrant images, where y_i is a binary variable indicating spray deposits in the image or not. Let $g(y_i)$ be a function that assigns data to its stratum:

$$g(y_i) = \begin{cases} 1 & \text{if } y_i = 1 \\ 2 & \text{if } y_i = 0 \end{cases} \tag{1}$$

After using the function $g(y_i)$, we numerically sort each stratum based on filenames. Feature-based sampling can then be completed by taking the same random index from both strata. Since images were taken before and after spraying the only differences are the spray deposits.

Random sampling with a random data split and random stratified sampling in Sect. 5 is used to compare against feature-based stratified sampling. Random sampling with a random data split refers to randomly splitting the data into training, validation, and test to then sample randomly. With random stratified sampling, the data are stratified using our methodology but sampled randomly without feature-based sampling.

3.2 Convolutional neural network architectures

Experiments with four pre-trained CNNs, DenseNet121 [40], EfficientNetB0 [41], ResNet18 [42], and VGG11 [43], have been completed. Previously, we found success with these networks [2, 3] and will continue to use them. All networks are pre-trained on the ImageNet [44] dataset and are then fine-tuned on our dataset allowing for prior

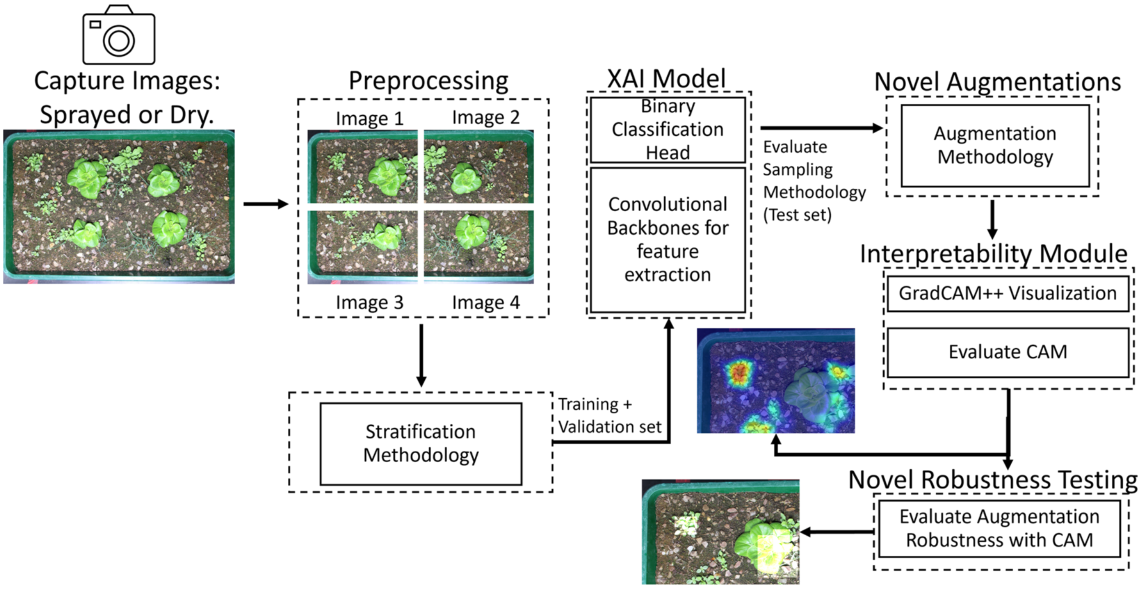

Capture Images: Sprayed or Dry.
Preprocessing
Image 1
Image 2
Image 3
Image 4
Stratification Methodology
Training + Validation set
XAI Model
Binary Classification Head
Convolutional Backbones for feature extraction
Evaluate Sampling Methodology (Test set)
Novel Augmentations
Augmentation Methodology
Interpretability Module
GradCAM++ Visualization
Evaluate CAM
Novel Robustness Testing
Evaluate Augmentation Robustness with CAM

Fig. 2 XAI pipeline with robustness testing

knowledge, faster training times, and improved accuracy [4].

All CNNs are given the same learning parameters during the training phase, to identify which CNN is the most effective without hyperparameter tuning. We resize our images to 864 x 1296 pixels as spray deposits are incredibly small; therefore, if images are resized to smaller dimensions, this can lead to information loss which makes it much more difficult to locate spray deposits. Therefore, we do not resize images to be any smaller. In the pipeline, a stochastic gradient descent optimiser with a learning rate of 0.001, a momentum of 0.9, and a weight decay of 0.1 has been used. Additionally, a learning rate scheduler that decays the learning rate, with a step size of 5, and a gamma constant of 0.3 was used. We train for a maximum of 30 epochs.

3.3 Classification metrics

To evaluate our pipeline in terms of classification, as this is a binary-class dataset, F1-score and area under the receiver operating characteristic (AUROC) are used. We define precision and recall to explain F1-score, and we define the false-positive rate and the true-positive rate to explain AUROC.

True positive ($\mathrm{T_P}$) is the number of images classified correctly as positive (sprayed); true negative ($\mathrm{T_N}$) is when a class is correctly predicted as negative (dry); false positive ($\mathrm{F_P}$) is counted when the network incorrectly predicts another class (misclassification); and false negative ($\mathrm{F_N}$) is counted when the network incorrectly predicts the positive class as negative (missed classification). Based on those, we can define the precision, recall, and F1-score values of a network using the following equations.

- Recall and Precision: Recall measures the networks' ability to detect positive samples, whereas precision measures the networks' accuracy in classifying a sample as positive. So, for example, precision measures, out of all of the sprayed images that the model predicted as sprayed, how many were sprayed, and recall measures, out of the images that were sprayed, how many did the model predict correctly. These can be defined as follows:

$$\mathrm{Recall} = \frac{\mathrm{T_P}}{\mathrm{T_P} + \mathrm{F_N}}. \tag{2}$$

$$\mathrm{Precision} = \frac{\mathrm{T_P}}{\mathrm{T_P} + \mathrm{F_P}}. \tag{3}$$

- F1-Score: Combines the precision and recall of a network by taking their harmonic mean. As we would like to optimise both precision and recall, F1 is the better metric. This metric is primarily used to compare the performance of multiple networks. F1-score can be defined as follows:

$$F1\text{-Score} = 2 \times \frac{(\mathrm{Precision} \times Recall)}{(\mathrm{Precision} + \mathrm{Recall})} \tag{4}$$

- FPR: False-positive rate (FPR) is a measure of the proportion of negative instances that are incorrectly classified as positive. FPR can be defined as follows:

$$\mathrm{FPR} = \frac{\mathrm{F_P}}{(\mathrm{T_N} + \mathrm{F_P})} \quad (5)$$

- TPR: True-positive rate (TPR) is also known as recall, see Eq. 2.
- AUROC: AUROC is defined as the area under the receiver operating characteristic (ROC) curve. The ROC curve is created by plotting the TPR against the FPR at various threshold settings for a binary classifier. Where 1 indicates a perfect performance and 0.5 indicates the performance of a random classifier. AUROC can be defined as follows:

$$\mathrm{AUROC} = \int_0^1 \mathrm{TPR}(\mathrm{FPR})\mathrm{dFPR} \quad (6)$$

3.4 CAM metrics

We will evaluate the patterns identified in the CAMs using deletion, insertion, and WSOD. Instead of weakly supervised object localisation (WSOL), we use WSOD because localisation typically pertains to a single object instance. In our case, we have multiple instances of spray deposits in each image of our dataset. We exclusively tested the sprayed images for all CAM metrics as we aim to detect spray deposits in a weakly supervised manner. We reported the average score across the entire test set in Sect. 5.

For deletion and insertion, the confidence, which is the probability of a given class, from a CNN prediction is recorded. Using the highest importance areas from a CAM, we remove or insert those regions increasing the area by 1% for both until the entire image is deleted or inserted. After plotting the confidence values from the CNN against the amount of each image that is deleted or inserted, the area under the curve (AUC) is calculated using the trapezoidal rule as follows:

$$\mathrm{AUC} = \frac{h}{2}[y_0 + 2(y_1 + y_2 + y_3 + \cdots + y_{n-1}) + y_n] \quad (7)$$

where y is the prediction confidence, n is equal to the number of plotted points, and h is equal to the increase in deletion or insertion change. Therefore, deletion scores that are lower are better and insertion scores that are higher are better.

WSOD will be completed with Intersection over Union (IoU). IoU can be calculated as the area of overlap divided by the area of union using the prediction from the CAM and ground truth bounding box. Mathematically, it is defined as follows:

$$J(A, B) = \frac{A \cap B}{A \cup B} \quad (8)$$

where A and B are the prediction and ground truth bounding boxes, respectively.

3.5 Assessing robustness

Using the CAMs as a starting point, we can also assess the CNN's robustness when images are augmented. Here, robustness refers to the CNN's ability to detect changes in an image that affects the confidence of its class prediction. We can augment test images to identify patterns that may be used for CNN prediction. We will take the highest importance region and augment that region first before increasing the area by 1% until the entire image is augmented. An average, using the entire test set, AUC will be recorded using Eq. 7 and reported in Sect. 5. In all augmentation scenarios, a higher AUC score indicates the CNN's greater capacity to predict in the presence of that augmentation. This also allows us to identify which features after various filtering are vital for classification. For instance, if an augmentation enhances edges and results in a larger AUC compared to other augmentations, it signifies that the edges of the image play a crucial role in prediction.

The augmentation techniques employed in this research are equalisation, flipping and mirroring, inversion, increasing and decreasing brightness, increasing and decreasing contrast, blurring, and sharpening. We chose these augmentations because they are widely used in the literature and are readily available in most public libraries. Each of the augmentation methods is described below:

- Equalisation (1 in Table 8): Equalise the image histogram. This function applies a nonlinear mapping to the input image, in order to create a uniform distribution of values in the output image.
- Flipping and mirroring (2 in Table 8): Flip image horizontally (left to right) then flip the image vertically (top to bottom).
- Inversion (3 in Table 8): Negate the image.
- Brightness (4 and 5 in Table 8): Increase the brightness by a factor of 1.2 or decrease the brightness by a factor of 0.8.
- Contrast (6 and 7 in Table 8): Increase the contrast by a factor of 1.2 or decrease the contrast by a factor of 0.8.
- Blur (8 in Table 8): Blur the image with a Gaussian blur.
- Sharp (9 in Table 8): Sharpen the image by a factor of 2.

The robustness testing pipeline is shown in Fig. 3 as a visual example, where we used "contrast up" as an example. We start with the quadrant image and using the CAM from the example we apply the mentioned augmentation to that region. Then, we test the CNN's confidence and create a confidence plot against the degree of image augmentation until the entire image is modified, similar to how deletion and insertion are plotted. Once this process is done, we calculate the AUC and report the average across the entire test set in Sect. 5.

4 Dataset description

Precision spraying for this study was completed with the XY spot spraying system, depicted in Fig. 4. The system uses a gantry system to move a spray plate to locations to be sprayed. The spray height can be changed through the usage of a removable floor. A Canon 500D camera is attached to the system to capture images. The spraying height from the spray plate to the tray bed is 30 cm whilst the distance from the camera lens to the tray bed is 45 cm. Spray deposits were completed with a pressure of 3 bar with a spray time of 8 ms which was recommended by Syngenta.

4.1 Lettuce trays

To create a near-realistic setting, our industrial partner Syngenta provided us with partially grown lettuces in trays with even spacing, and the trays also had commonly found weeds sown randomly in different parts of the tray. In Table 1, we have included the Biologische Bundesanstalt, Bundessortenamt und CHemische Industrie (BBCH) breakdown for the plants and weeds used in our experimental scenario.

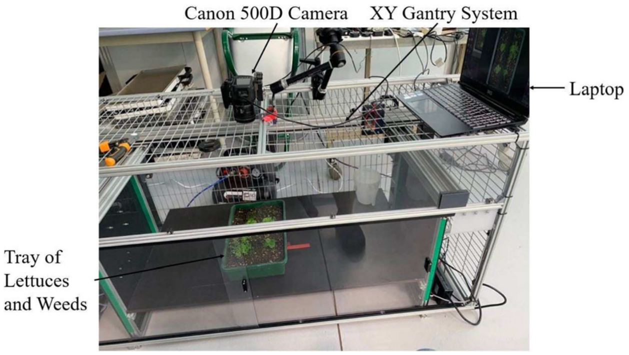


Fig. 4 XY sprayer with tray of lettuces and weeds

4.2 Data collection process

For the data collection process, we placed a Canon 500D camera on top of the XY sprayer system with the tray placed underneath (shown in Fig. 4) so that the camera could clearly capture each tray including the corners of the tray. The system was then operated and controlled to target and spray on each weed region once. Figure 5 shows a comparison of the same tray before and after spraying with a specific region of weeds zoomed in. As can be seen from these images, it is a difficult task to differentiate the original images as the spray deposits are very small.

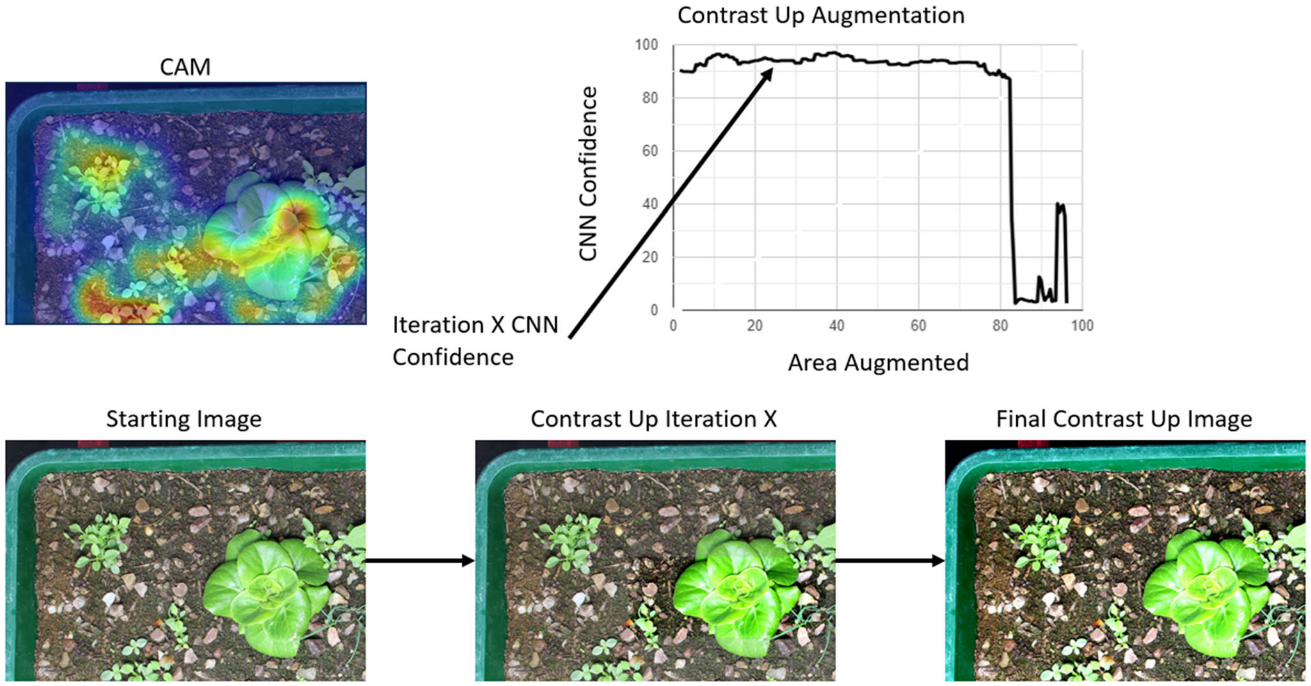


Fig. 3 Robustness testing pipeline with contrast up as an example

4.3 Slicing-aided hyper inference (SAHI) and relabelling

We expanded our dataset size by splitting each image using a method called SAHI proposed in 2022 by Akyon et al. [36]. Initially, images in our dataset included trays with four different crops and randomly scattered weeds. To make the dataset more representative of commercial farms where systems move over rows with a single crop per row, we divided each image into four quadrants, each containing one crop. This process resulted in images that closely resemble what is typically encountered in commercial farming practices. An illustration of SAHI process we employed in our dataset is shown in Fig. 6.

Now that each image is divided into quadrants, we must label each quadrant as either sprayed or dry. In Fig. 7, you can see that whilst some quadrants were initially part of a sprayed tray, certain individual quadrants may not have any spray deposits in them.

4.4 Novel data augmentation methods

To enhance the learning process for the CNNs for this problem, we have devised two unique image augmentation methods. We employed a subtraction-based and a threshold-based augmentation stage which were specifically designed in consultation with the domain experts to aid in the spray detection process and improve the model performance matrices detailed in Sect. 4.

4.4.1 Subtracted augmentation

Our dataset comprises images captured before and after spraying. By subtracting one from the other, we can precisely identify pixel-wise differences between the two images. We then add this difference to the original image, intensifying areas of change. To ensure quality, we filter out noise by ignoring insignificant changes in small areas. An example demonstrating this process is shown in Fig. 8. Subtracting the dry image from the sprayed image and adding the result back to the sprayed image yields our new novel image. This results in subtle yet precise increases in intensity in areas of interest.

This can be formalised as follows:

$$s(I) = \begin{cases} I_{2s} = I_s + (I_s - I) & \text{if } g(y_i) = 1 \\ I_s = I + (I - I_s) & \text{if } g(y_i) = 0 \end{cases} \tag{9}$$

where I is the input image that if contains spray deposits creates case 1 and if not creates case 2.

However, as shown in Fig. 8, there are instances of spray deposits on the tray that miss targets, which go unnoticed by the subtraction method. One such missed region is shown using the red bounding box. The subtraction augmentation relies on the assumption that targets sprayed will move due to the force of the spray actuation. However, as the tray remained stationary, no information was available about where the subtraction had taken place. Such movement may not occur for larger targets, as the force from the actuation may be insufficient to displace them.

4.4.2 Thresholding augmentation

Therefore, we tested a new augmentation method based on colour thresholding. We employ colour thresholding to identify crops and weeds, subsequently enhancing the intensities in regions relevant to where the CNN should focus. This enhanced image is then combined with the original, as illustrated in Fig. 9. The result is an image with increased intensity in areas of interest, potentially containing spray deposits on both the targets and non-targets, given the significance of these regions.

We also tested a combination of both augmentations. Images generated from these augmentation processes contributed to the final dataset including the quadrant image, and images from subtracted augmentation, and thresholding augmentation. We hypothesised, the combination would bring the benefits of both augmentations for the developed CNNs.

Table 1 BBCH scale for plants used

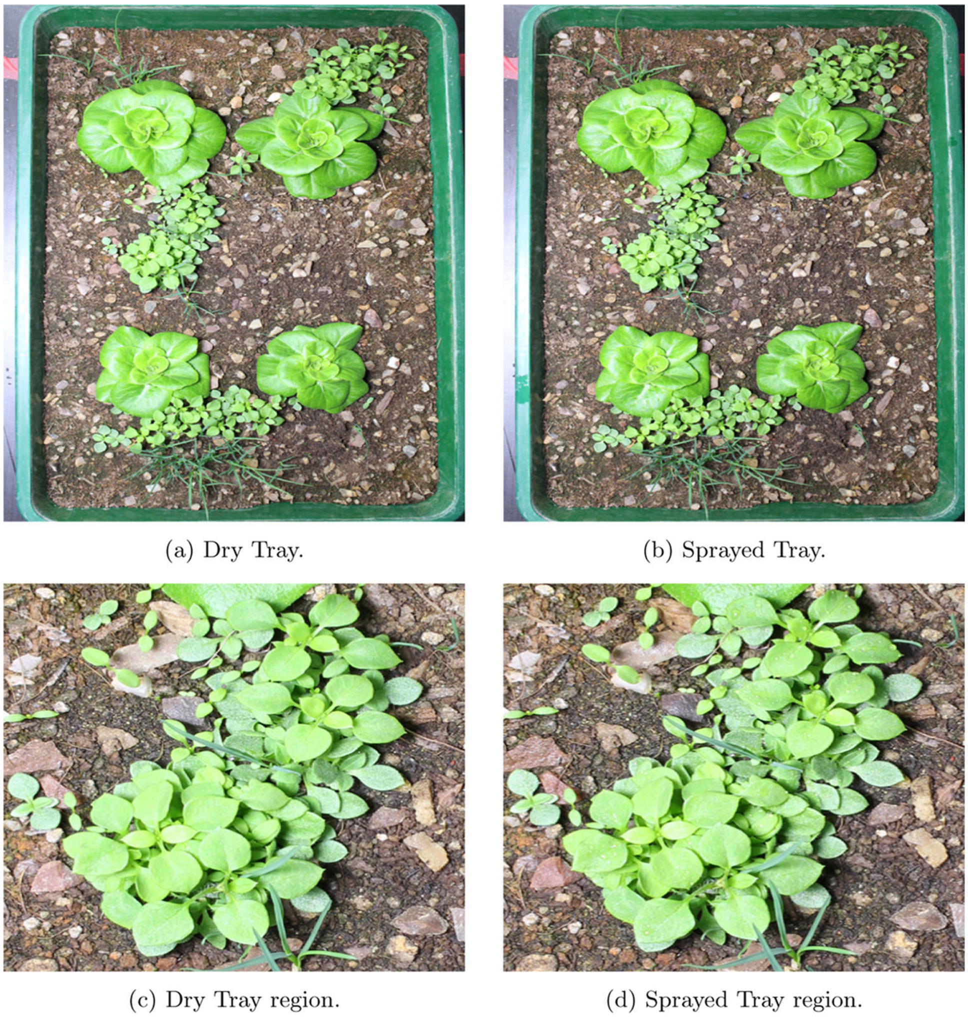
Latin name	EPPO code	Common name	BBCH
Poa annua	POAAN	Annual Meadowgrass	10–13
Stellaria media	STEME	Chickweed	10–22
Lactuca sativa	LACSA	Lettuce	19

5 Results and performance evaluation

For ease of reading, we split our results into sampling evaluation, considering classification metrics, and novel augmentation evaluation considering classification, CAM, and our novel robustness methodology metrics. Each section has the original data split and the tenfold cross-validation evaluation.

5.1 Sampling evaluation

In each table within in Sect. 5, the best results for each metric are shown in bold for each CNN. Table 2 shows the results for the comparison of the sampling methods. In all


(a) Dry Tray.

(b) Sprayed Tray.

(c) Dry Tray region.

(d) Sprayed Tray region.

Fig. 5 Comparison of dry tray **a**, sprayed tray **b**, as well as the chickweed region in both dry **c** and sprayed **d**

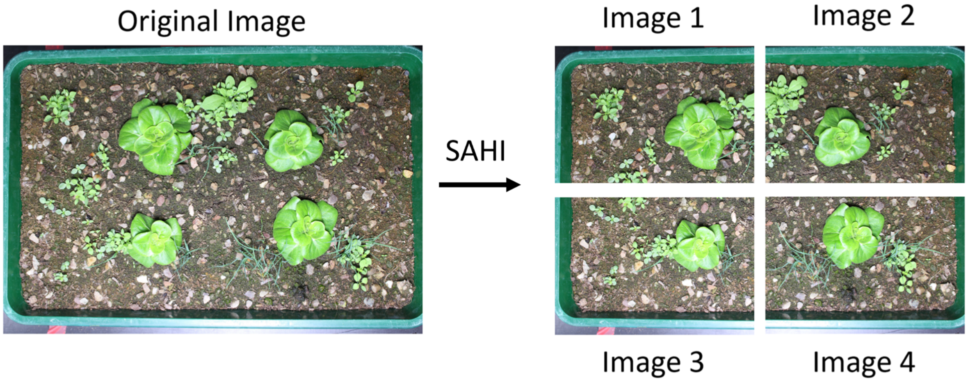


Fig. 6 Slicing-aided hyper inference (SAHI) process applied on the collected data

architectures, the usage of feature-based stratified sampling is best across all classification metrics used. When using feature-based stratified sampling, the largest increase from the random sampling and random data split is 18.09%, 17.41%, and 17.74%, for F1 dry, F1 sprayed, and AUROC, respectively, for the VGG11.

When considering the tenfold cross-validation classification scores, the effect of feature-based stratified sampling is much larger as shown in Table 3. The standard deviation shown after the average score shows that all folds when considering the feature-based stratified sampling are very small, within 3.07% across the EfficientNetB0, DenseNet121, and ResNet18. This shows that the method is consistent across the data regardless of how the data are splitted for training and testing. All models tested without feature-based stratified sampling are unable to score higher

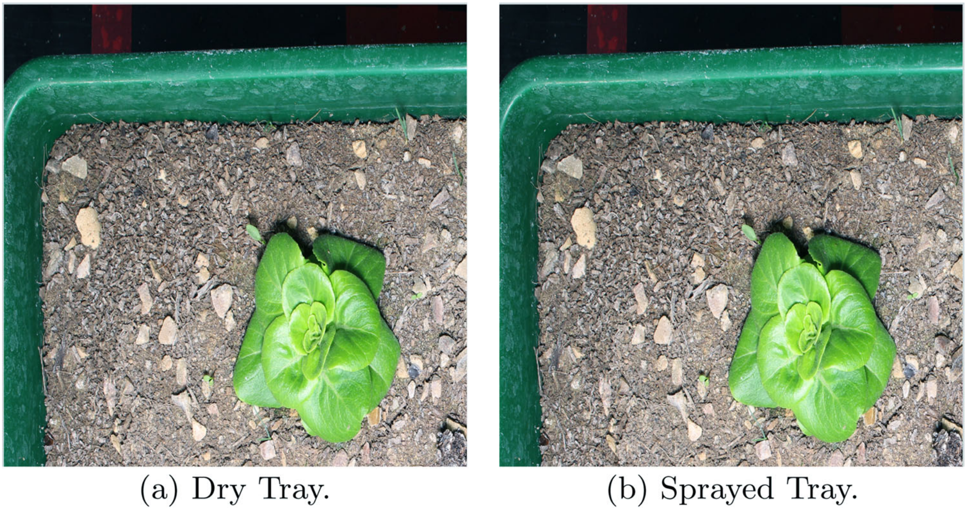


Fig. 7 Dry tray quadrant **a** against sprayed tray quadrant **b** where sprayed tray quadrant has no spray deposits

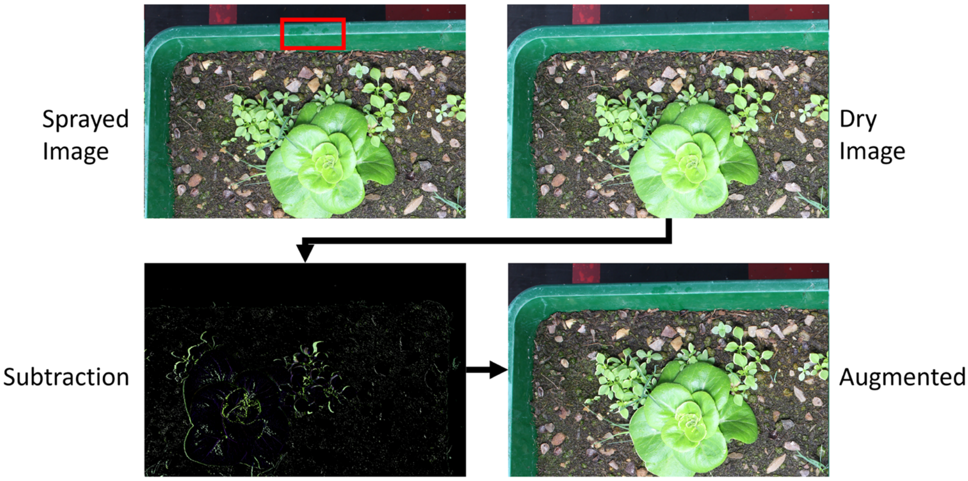


Fig. 8 Subtraction augmentation example, with instances of spray deposits on the tray with the missed target (the missed region is shown using the red bounding box)

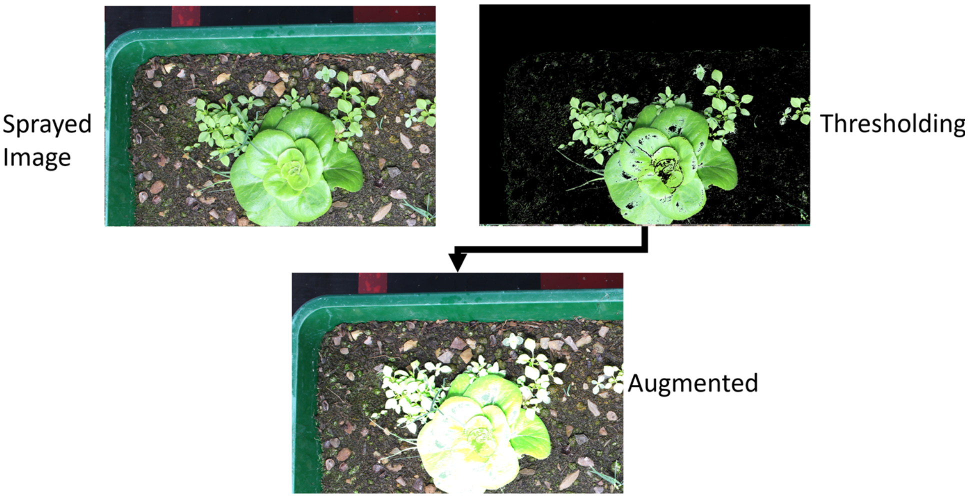


Fig. 9 Thresholding augmentation example

than 52% AUROC, which is essentially a random classifier. Whereas, with the proposed sampling methodology, the minimum AUROC is 84.98% with the VGG11. The VGG11 has a significant decrease when comparing back to the original data split. This is likely due to the VGG11 model being the smallest of the models tested with far less convolutional layers. The EfficientNetB0 and DenseNet121 AUROC scores improve from the original split. The ResNet18 across all metrics is within 4% of the original data split. From these results, we will use feature-based stratified sampling as the sampling and data split method for our augmentations following the same training loop.

5.2 Novel augmentation evaluation

When we analyse the proposed augmentations, we observe that the majority of the results match the baseline or exhibit improvements across all classification metrics in the original data split. In Table 4, the subtraction augmentation consistently yields the best performance across almost all architectures. The EfficientNetB0 architecture demonstrates improvement across all metrics when this augmentation is applied. DenseNet121 matches the classification metrics with subtraction and both augmentations when combined. The ResNet18 performs equally well under all metrics either with thresholding or subtraction used separately. The subtraction and thresholding augmentation performs best when using the VGG11 architecture; however, more testing is needed to fully understand why the classification scores drop by 9% for the VGG11 model when comparing back to Table 2. However, as mentioned with the tenfold cross-validation classification scores, it is likely due to the model size and potentially a lower complexity when compared to the other models tested.

Table 5 shows the cross-validated results when applying the proposed augmentations to the appropriate folds. It can be shown that the EfficientNetB0 architecture improves only in the subtraction augmentation across all metrics when compared to the original tenfold cross-validation in Table 3. The other augmentations are within 2% across all metrics for the EfficientNetB0. DenseNet121 is within 2% when using the thresholding augmentation when considering all metrics. The ResNet18 is able to improve with the subtraction augmentation as well as using both subtraction and thresholding. However, similarly to the original data split, the VGG11 struggles with augmentations and has a large drop. This is likely due to the limited architecture size as it is significantly smaller than the other architectures with far less layers.

In Table 6, deletion, insertion, and WSOD scores are reported when using the augmentations and feature-based stratified sampling against the baseline of no augmentations. From the results, the architecture with the best WSOD is the VGG11 with 42.78% when using no augmentation. Further analysis is needed to fully understand why this occurs. The VGG11 model does not act similarly to any of the other models that have been developed, and this could be due to its lower complexity. However, the most effective augmentation when using WSOD is subtraction as two of the four models tested have the highest scores from this augmentation. The explanations from all augmentations tested are not exactly causal in any case, despite deletion and insertion being designed to complement each other. However, all augmentations on all models, apart from VGG11, have a lower deletion than insertion on average, and therefore, on average, on the original data split generate CAMs that are representations of regions of interest that each model is using for classification. Deletion when not using augmentations scores the lowest across all CNNs tested. Insertion when using subtraction scores the highest across all CNNs tested.

When completing a cross-validated approach for the CAM results, WSOD has to be excluded as each fold would need to be labelled defeating the point of WSOD. Table 7, therefore, shows the results for deletion and insertion with the augmented folds. The results show that, on average, adding the augmentation makes it easier to

Table 2 Sampling data split method results

Architecture	Data split method	*F*1 dry (%)	*F*1 sprayed (%)	AUROC (%)
EfficientNetB0	Random sampling random split	92.30	91.52	91.94
	Random stratified sampling	**93.94**	**93.10**	**93.55**
	Feature-based stratified sampling	**93.94**	**93.10**	**93.55**
DenseNet121	Random sampling random split	93.75	93.33	93.55
	Random stratified sampling	93.75	93.33	93.55
	Feature-based stratified sampling	**95.24**	**95.08**	**95.16**
ResNet18	Random sampling random split	87.09	87.09	87.10
	Random stratified sampling	95.38	94.91	95.16
	Feature-based stratified sampling	**96.88**	**96.67**	**96.77**
VGG11	Random sampling random split	78.68	79.36	79.03
	Random stratified sampling	91.52	92.30	91.94
	Feature-based stratified sampling	**96.77**	**96.77**	**96.77**

Bold values indicate the best performance for each architecture

Table 3 Tenfold cross-validation sampling data split method results

Architecture	Data split method	$F1$ dry (%)	$F1$ sprayed (%)	AUROC (%)
EfficientNetB0	Random sampling random split	22.73 ± 28.95	54.55 ± 20.55	52.00 ± 2.04
	Random stratified sampling	15.50 ± 25.78	56.39 ± 21.25	50.86 ± 0.96
	Feature-based stratified sampling	**94.16 ± 2.45**	**93.39 ± 3.07**	**93.87 ± 2.73**
DenseNet121	Random sampling random split	0.00 ± 0.00	66.66 ± 0.00	50.00 ± 0.00
	Random stratified sampling	0.00 ± 0.00	66.66 ± 0.00	50.00 ± 0.00
	Feature-based stratified sampling	**95.48 ± 1.20**	**95.28 ± 1.30**	**95.39 ± 1.24**
ResNet18	Random sampling random split	0.54 ± 1.62	66.02 ± 0.19	50.00 ± 0.00
	Random stratified sampling	0.00 ± 0.00	66.66 ± 0.00	50.00 ± 0.00
	Feature-based stratified sampling	**93.94 ± 2.28**	**93.32 ± 2.81**	**93.65 ± 2.52**
VGG11	Random sampling random split	6.66 ± 19.99	60.00 ± 19.99	50.00 ± 0.00
	Random stratified sampling	13.33 ± 26.66	53.33 ± 26.66	50.00 ± 0.00
	Feature-based stratified sampling	**85.25 ± 8.61**	**83.52 ± 11.68**	**84.98 ± 8.21**

Bold values indicate the best performance for each architecture

Table 4 Augmentation (feature-based stratified sampling) classification results

Architecture	Augmentation	$F1$ dry (%)	$F1$ sprayed (%)	AUROC (%)
EfficientNetB0	Thresholding	93.93	93.10	93.55
	Subtraction	**95.38**	**94.91**	**95.16**
	Subtraction and thresholding	89.23	88.13	88.71
DenseNet121	Thresholding	91.80	92.06	91.94
	Subtraction	**95.24**	**95.08**	**95.16**
	Subtraction and thresholding	**95.24**	**95.08**	**95.16**
ResNet18	Thresholding	**93.54**	**93.54**	**93.55**
	Subtraction	**93.54**	**93.54**	**93.55**
	Subtraction and thresholding	93.75	93.33	93.55
VGG11	Thresholding	77.92	63.82	72.58
	Subtraction	87.32	83.01	85.48
	Subtraction and thresholding	**85.71**	**88.23**	**87.10**

Bold values indicate the best performance for each architecture

Table 5 Tenfold cross-validation augmentation (feature-based stratified sampling) classification results

Architecture	Augmentation	$F1$ dry (%)	$F1$ sprayed (%)	AUROC (%)
EfficientNetB0	Thresholding	92.64 ± 1.91	91.70 ± 2.5	92.21 ± 2.16
	Subtraction	**94.7 ± 2.15**	**94.3 ± 2.47**	**94.52 ± 2.29**
	Subtraction and thresholding	92.42 ± 2.73	91.69 ± 3.08	92.07 ± 2.89
DenseNet121	Thresholding	**93.92 ± 1.45**	**93.65 ± 1.79**	**93.80 ± 1.60**
	Subtraction	91.78 ± 4.82	89.85 ± 9.04	91.04 ± 6.34
	Subtraction and thresholding	91.58 ± 2.70	90.71 ± 3.66	92.03 ± 3.11
ResNet18	Thresholding	93.68 ± 2.17	92.29 ± 2.79	93.36 ± 2.44
	Subtraction	**94.06 ± 2.30**	**93.81 ± 2.47**	**93.95 ± 2.37**
	Subtraction and thresholding	**94.40 ± 2.22**	**94.05 ± 2.40**	**94.23 ± 2.30**
VGG11	Thresholding	**77.48 ± 6.69**	**75.51 ± 9.13**	**77.06 ± 6.34**
	Subtraction	65.52 ± 15.11	72.56 ± 7.15	71.17 ± 6.75
	Subtraction and thresholding	70.39 ± 12.02	65.85 ± 9.70	69.35 ± 8.43

Bold values indicate the best performance for each architecture

Table 6 Augmentation (feature-based stratified sampling) CAM results

Architecture	Augmentation	Deletion (%)	Insertion (%)	WSOD (%)
EfficientNetB0	None	**20.38**	19.41	**37.22**
	Thresholding	65.96	93.02	34.34
	Subtraction	82.05	**97.53**	36.70
	Subtraction and thresholding	68.45	89.70	30.09
DenseNet121	None	**14.20**	2.37	**35.44**
	Thresholding	49.70	93.48	27.71
	Subtraction	52.34	**95.90**	28.39
	Subtraction and thresholding	79.42	87.67	34.47
ResNet18	None	**4.65**	2.91	**38.75**
	Thresholding	73.55	79.16	32.62
	Subtraction	88.07	**92.30**	37.53
	Subtraction and thresholding	74.55	84.37	32.81
VGG11	None	**3.29**	7.64	**42.78**
	Thresholding	55.22	43.61	35.94
	Subtraction	70.11	**65.77**	35.69
	Subtraction and thresholding	70.53	64.15	35.33

Bold values indicate the best performance for each architecture

create CAMs that are explainable as deletion is lower than insertion regardless of augmentation methodology. Furthermore, it shows without the augmentations the CAMs are not explainable as all models have a higher deletion than insertion.

5.2.1 Assessing robustness with augmentations

For a visual comparison on the effect of augmentations, Fig. 10 shows one of the test images with CAMs from each CNN tested. Figure 10 shows the ground truth with bounding boxes against no augmentation, subtraction, thresholding, and using both augmentations. At the top of the image is the ground truth for spray deposits, on the first (top) row is the EfficientNetB0 CAM, on the second row is the DenseNet121 CAM, on the third row is the ResNet18 CAM, and on the fourth (bottom) row is the VGG11 CAM. As shown from Fig. 10, the addition of augmentations does change the CAM. When using no augmentation, the CAMs are able to locate spray deposits in a WSOD task as reported in Table 6. With the addition of the subtraction augmentation area, intensities increase in all CNNs except the VGG11. The ResNet18 has the best improvement visually as it has more areas that overlap with the ground truth. When looking at thresholding, the effect also increases interest in the thresholding objects for all architectures tested. Finally, when using both augmentations, the intensities are increased for regions where spray deposits are located.

Table 8 presents the robustness results of our proposed augmentation methods. Subtraction and both augmentations consistently outperform the non-augmented versions. We first report the increase from the highest baseline (no augmentations) score and then the maximum increase from the baseline:

- Equalisation (1) with the EfficientNetB0 without augmentation scores 18.70% and improves by 58.73% with Thresholding. ResNet18 sees a dramatic 86.98% improvement with subtraction.
- Flipping and mirroring (2) records a maximum baseline of 29.09% with ResNet18, thresholding raises the score by 47.11%. Whilst DenseNet121 experiences a 92.44% boost with subtraction.
- Inversion (3) with the EfficientNetB0 begins with 20.40%, then increases by 37.69% with both augmentations. ResNet18 scores increase by 84.6% using subtraction.
- Brightness increase (4) scores highest with ResNet18 at 32.27% with no augmentation then increases by 48.69% with both augmentations. Subtraction achieves an increase of 88.61% with the DenseNet121.
- Brightness decrease (5) sees its best baseline with EfficientNetB0 18.92%. Thresholding usage increases the score by 58.34%. ResNet18 has the largest increase with 87.46% when using subtraction.
- Contrast increase (6) records the highest score with ResNet18 at 31.37% without augmentation. Thresholding and Subtraction boost scores by 69.28% and 91.26%, respectively.
- Contrast decrease (7), with EfficientNetB0, starts at 19.92% and sees improvement of 53.79% using both

Table 7 Tenfold cross-validation augmentation (feature-based stratified sampling) CAM results

Architecture	Augmentation	Deletion (%)	Insertion (%)
EfficientNetB0	None	54.61 ± 27.76	47.59 ± 14.06
	Thresholding	**55.62 ± 32.80**	**56.52 ± 33.47**
	Subtraction	**54.01 ± 31.30**	**56.03 ± 28.22**
	Subtraction and thresholding	**41.60 ± 16.27**	**56.10 ± 27.72**
DenseNet121	None	49.38 ± 18.32	48.06 ± 17.31
	Thresholding	**56.80 ± 26.92**	**59.07 ± 24.64**
	Subtraction	**49.06 ± 29.97**	**52.67 ± 29.54**
	Subtraction and thresholding	**46.34 ± 18.73**	**51.20 ± 17.44**
ResNet18	None	50.84 ± 23.82	50.50 ± 28.87
	Thresholding	**50.48 ± 31.79**	**52.55 ± 33.37**
	Subtraction	**44.21 ± 21.33**	**46.44 ± 20.72**
	Subtraction and thresholding	**47.55 ± 24.61**	**53.82 ± 28.82**
VGG11	None	49.76 ± 14.79	46.63 ± 17.10
	Thresholding	**45.88 ± 17.42**	**47.18 ± 18.79**
	Subtraction	**47.77 ± 21.95**	**48.13 ± 21.04**
	Subtraction and thresholding	**40.79 ± 21.93**	**49.12 ± 14.18**

Bold values indicate performance where Deletion is lower than Insertion

augmentations. ResNet18 experiences an 80.1% jump with subtraction.

- Blurring (8), using EfficientNetB0, has a baseline of 19.34% and an increase of 38.46% with thresholding. The improvement of 58.16% is observed with DenseNet121 using both augmentations.
- Sharpening (9) starts highest with ResNet18 at 55.98% that increases by 19.74% with thresholding. With both augmentations applied to EfficientNetB0, improvements of 47.48% are made.

In our robustness experiments prior to applying novel augmentation methods, we found that three kernels sharpening, brightness, and contrast increasing consistently improved CNN detection. This discovery aligns with our common understanding. Sharpening, for instance, sharpens the edges of spray deposits, resulting in clearer outlines. According to our findings, this kernel is consistently employed across all tested CNNs. The second most frequently used kernel is relevant to brightness adjustment, which naturally enhances the luminosity of the deposits, making them easier to identify. Similarly, enhancing contrast brings out the differences between light and dark areas, further enhancing the visibility of spray deposits. This makes it the third most commonly used kernel across the tested CNNs in the absence of any additional augmentation in the input data.

With the augmented data, the top three kernels used can also be monitored for each augmentation. Specifically, the thresholding augmentation exhibits the highest reliance on three key filtering kernels: flipping and mirroring, contrast, and brightness increasing kernels. The use of thresholding augmentation introduces intensities at specific spatial locations, encouraging CNNs to assign greater importance to these regions. Consequently, CNNs achieve their highest average scores when exposed to spatial transformations such as flipping and mirroring, regardless of the architecture. This observation is further supported by the significance of contrast increase, which emphasises distinct regions that have undergone thresholding. Additionally, brightness increase aligns with these findings, as it visually enhances intensities within regions, creating a sense of brightness amplification. Collectively, these results highlight the consistent and crucial role of these top three kernels across various augmented datasets.

In the context of the subtraction augmentation, the top three filtering kernels, on average across all CNNs, consist of sharpening, flipping and mirroring, and contrast increase. The subtraction augmentation increased pixel intensity to localised areas of the images, highlighting regions where crops or weeds have shifted either before or after spraying. The higher scores for this augmentation can be explained by the fact that it enhances fine and granular regions compared to thresholding. In this case, subtraction creates fine lines tracing the edges of individual crops or weeds affected by movement. This visual similarity to the sharpening process clarifies its top position in average scores in this scenario. Similar to thresholding augmentation, flipping and mirroring, along with contrast increase, maintain their positions among the top three filtering kernels due to their impact on intensifying specific spatial areas.

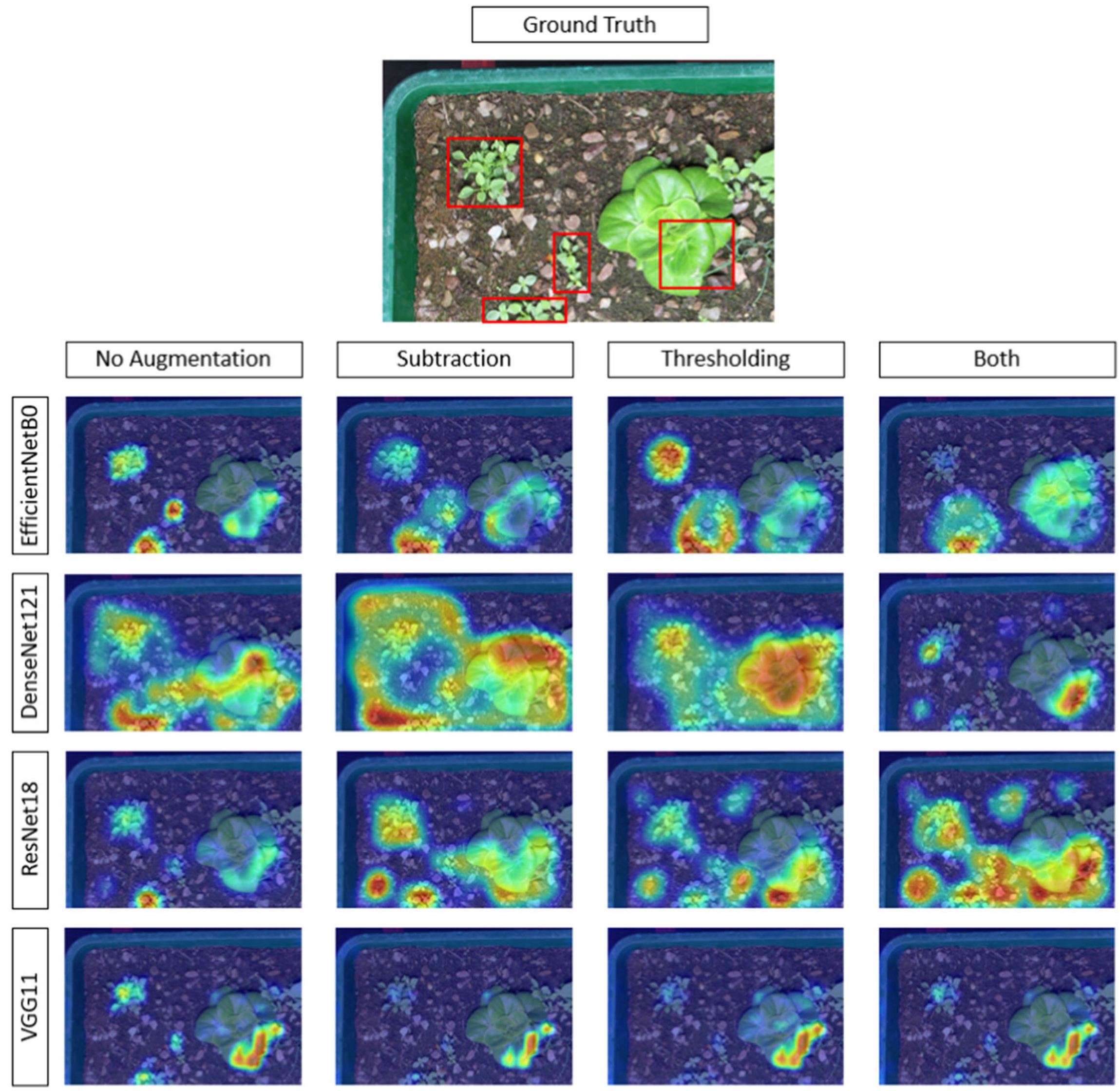


Fig. 10 Comparison of CAMs considering augmentations against labelled bounding box ground truth image

Using both augmentations in the training dataset, the highest-scoring top three filtering kernels are, flipping and mirroring, sharpening, and contrast increase. This makes sense since the combination of both augmentations enables the models to learn both spatial importance and small detail differentiation as previously stated.

When applying the tenfold cross-validation for robustness testing, as there are a large number of results, we have taken the average across all tests and plotted this with the average standard deviation in Fig. 11. Visually this reinforces the points made in the original data split. Even when considering the highest scores across all splits of data for no augmentation, this does not meet the baselines of the lowest scoring augmented results for any of the models tested.

6 Conclusion

In summary, the proposed approach significantly improves precision and efficiency in spray deposit detection. Notably, we achieve these improvements without the need for costly and cumbersome tracers or WSPs, which are traditionally used in agricultural field settings. We eliminated the reliance on these materials in our detection process by using images only. Our method not only streamlines the process but also reduces the associated costs for monitoring the environmental impact, thus contributing to sustainable and economically viable agricultural practices.

Additionally, our methodology excels in classification and detection tasks, accurately pinpointing spray deposit locations within images. What distinguishes our approach is its ability to achieve these results without requiring to

Table 8 Robustness results

Architecture	Augmentation	1 (%)	2 (%)	3 (%)	4 (%)	5 (%)	6 (%)	7 (%)	8 (%)	9 (%)
EfficientNetB0	None	18.70	11.91	20.40	14.75	18.92	12.65	19.92	19.34	43.75
	Thresholding	77.43	86.23	69.52	82.26	77.26	81.93	64.69	57.80	86.18
	Subtraction	**84.71**	90.14	**80.10**	**89.30**	84.93	**87.11**	**79.79**	**79.16**	90.85
	Both	78.24	**90.99**	58.09	82.08	**88.38**	83.48	70.28	40.63	**91.23**
DenseNet121	None	6.00	3.91	15.69	4.06	4.36	3.68	10.49	22.61	3.53
	Thresholding	84.45	94.42	52.11	90.57	85.58	93.96	36.99	50.08	93.81
	Subtraction	**91.59**	**96.35**	49.14	**92.67**	85.32	**94.94**	34.50	29.24	**95.52**
	Both	85.08	86.84	**79.43**	85.90	**86.62**	87.53	**84.09**	**80.77**	88.03
ResNet18	None	3.71	29.09	4.89	32.27	3.63	31.37	4.59	4.71	55.98
	Thresholding	76.63	76.20	75.22	76.84	76.18	77.70	73.02	74.21	75.72
	Subtraction	**90.69**	**92.00**	**89.49**	**91.29**	**91.09**	**92.25**	**84.69**	**86.93**	**92.70**
	Both	80.91	80.41	78.78	80.96	79.54	82.05	73.00	74.33	80.22
VGG11	None	2.93	2.68	3.31	2.68	2.92	2.63	2.95	3.09	2.61
	Thresholding	55.69	56.13	55.35	56.14	55.73	56.54	55.42	56.31	56.31
	Subtraction	70.73	**71.53**	70.16	**71.16**	70.82	**72.19**	70.66	70.65	**72.27**
	Both	**70.79**	71.25	**70.57**	71.13	**70.94**	71.50	**70.79**	**70.82**	71.50

Bold values indicate the best performance for each architecture

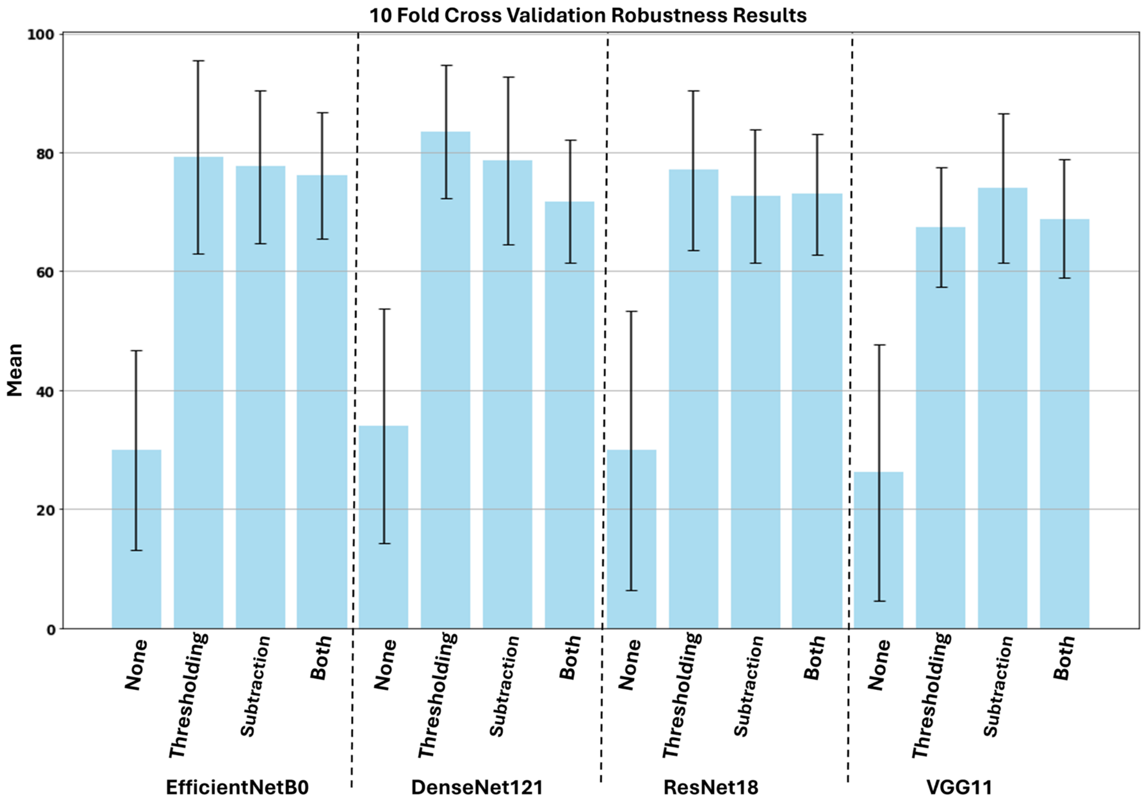


Fig. 11 Bar plot with standard deviation bars of the average across all tests

include an additional object detector stage in a WSOD process. This not only simplifies the implementation process but also makes it more computationally efficient, thus paving the way for widespread adoption in the Agri-Robotics field. We have also significantly enhanced the robustness of each CNN using novel domain-specific augmentations, resulting in a substantial improvement over the baseline. Furthermore, our robustness testing methodology helps us identify patterns within spatial regions crucial for CNN inference. For instance, we have discovered that brightness is a key indicator for identifying spray deposits in all tested CNNs, providing a foundation for deeper insights.

To deploy this type of pipeline for precision spraying, real-world data need to be collected. Therefore, automated systems will need to be changed to ensure images can be collected post-spraying or systems will need to be manually driven. Furthermore, systems that are already computationally overloaded will need to take further considerations to incorporate this methodology. The literature has shown that quantisation could be promising for resource-constrained devices, and XAI can be applied to these [45]. Finally, evaluating models with the robustness pipeline is computationally expensive and should be considered only where necessary.

In future, we plan to comprehensively compare our methodology with traditional evaluation methods on target weeds and non-target crops. We have plans to collect additional data with mobile spot spraying systems and develop methods for detecting challenging-to-identify spray deposits and also to improve the detection of the under-represented Annual Meadowgrass weeds.

Acknowledgements This work is supported by the Engineering and Physical Sciences Research Council [EP/S023917/1]. This work is also supported by Syngenta as the Industrial partner. The research presented in this paper was carried out on the High-Performance Computing Cluster supported by the Research and Specialist Computing Support service at the University of East Anglia.

Data availability The data currently are unavailable. • **Reason for Unavailability:** We would like to refine the dataset and then open source in future. The data were also collected on a proprietary system.

Declarations

Conflict of interest The authors declared that there is no conflict of interest.

References

1. Raja R, Slaughter DC, Fennimore SA, Siemens MC (2023) Real-time control of high-resolution micro-jet sprayer integrated with machine vision for precision weed control. Biosyst Eng 228:31–48. https://doi.org/10.1016/j.biosystemseng.2023.02.006
2. Rogers H, De La Iglesia B, Zebin T, Cielniak G, Magri B (2023) An agricultural precision sprayer deposit identification system. In: 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE), pp. 1–6. https://doi.org/10.1109/CASE56687.2023.10260374
3. Rogers H, De La Iglesia B, Zebin T, Cielniak G, Magri B (2023) An automated precision spraying evaluation system. In: Iida F, Maiolino P, Abdulali A, Wang M (eds) Towards Autonomous Robotic Systems. Springer, Cham, pp 26–37
4. Hasan ASMM, Sohel F, Diepeveen D, Laga H, Jones MGK (2021) A survey of deep learning techniques for weed detection from images. Comput Electron Agric 184:106067. https://doi.org/10.1016/j.compag.2021.106067
5. Wu Z, Chen Y, Zhao B, Kang X, Ding Y (2021) Review of weed detection methods based on computer vision. Sensors. https://doi.org/10.3390/s21113647
6. Liu B, Bruch R (2020) Weed detection for selective spraying: a review. Curr Robot Rep 1(1):19–26. https://doi.org/10.1007/s43154-020-00001-w
7. Ghiani L, Sassu A, Piccirilli D, Marcialis G, Gambella F (2020) Development of a Matlab Code for the Evaluation of Spray Distribution with Water-Sensitive Paper, pp. 845–853. https://doi.org/10.1007/978-3-030-39299-4_91
8. Kim J, Seol J, Lee S, Hong S-W, Son HI (2020). An intelligent spraying system with deep learning-based semantic segmentation of fruit trees in orchards. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 3923–3929 (2020). https://doi.org/10.1109/ICRA40945.2020.9197556
9. Cai J, Wang X, Gao Y, Yang S, Zhao C (2019) Design and performance evaluation of a variable-rate orchard sprayer based on a laser-scanning sensor. Int J Agric Biol Eng 12(6):51–57
10. Seol J, Kim J, Son HI (2022) Field evaluations of a deep learning-based intelligent spraying robot with flow control for pear orchards. Precis Agric 23(2):712–732. https://doi.org/10.1007/s11119-021-09856-1
11. Farooque AA, Hussain N, Schumann AW, Abbas F, Afzaal H, McKenzie-Gopsill A, Esau T, Zaman Q, Wang X (2023) Field evaluation of a deep learning-based smart variable-rate sprayer for targeted application of agrochemicals. Smart Agric Technol 3:100073
12. Wang B, Yan Y, Lan Y, Wang M, Bian Z (2023) Accurate detection and precision spraying of corn and weeds using the improved yolov5 model. IEEE Access 11:29868–29882. https://doi.org/10.1109/ACCESS.2023.3258439
13. Fu H, Zhao X, Wu H, Zheng S, Zheng K, Zhai C (2022) Design and experimental verification of the yolov5 model implanted with a transformer module for target-oriented spraying in cabbage farming. Agronomy. https://doi.org/10.3390/agronomy12102551
14. Gonzalez-de-Soto M, Emmi L, Perez-Ruiz M, Aguera J, Gonzalez-de-Santos P (2016) Autonomous systems for precise spraying – evaluation of a robotised patch sprayer. Biosystems Engineering 146, 165–182 https://doi.org/10.1016/j.biosystemseng.2015.12.018. Special Issue: Advances in Robotic Agriculture for Crops
15. Hanif AS, Han X, Yu S-H (2022) Independent control spraying system for uav-based precise variable sprayer: a review. Drones. https://doi.org/10.3390/drones6120383
16. Wang L, Song W, Lan Y, Wang H, Yue X, Yin X, Luo E, Zhang B, Lu Y, Tang Y (2021) A smart droplet detection approach with

vision sensing technique for agricultural aviation application. IEEE Sens J 21(16):17508–17516
17. Zheng K, Zhao X, Han C, He Y, Zhai C, Zhao C (2023) Design and experiment of an automatic row-oriented spraying system based on machine vision for early-stage maize corps. Agriculture 13(3):691
18. Liu L, Liu Y, He X, Liu W (2022) Precision variable-rate spraying robot by using single 3d lidar in orchards. Agronomy 12(10):2509
19. Gao S, Wang G, Zhou Y, Wang M, Yang D, Yuan H, Yan X (2019) Water-soluble food dye of allura red as a tracer to determine the spray deposition of pesticide on target crops. Pest Manag Sci 75(10):2592–2597
20. Raja R, Nguyen TT, Slaughter DC, Fennimore SA (2020) Real-time weed-crop classification and localisation technique for robotic weed control in lettuce. Biosys Eng 192:257–274
21. Liu J, Abbas I, Noor RS (2021) Development of deep learning-based variable rate agrochemical spraying system for targeted weeds control in strawberry crop. Agronomy 11(8):1480
22. Özlüoymak Barış (2022) Development and assessment of a novel camera-integrated spraying needle nozzle design for targeted micro-dose spraying in precision weed control. Comput Electron Agric 199:107134. https://doi.org/10.1016/j.compag.2022.107134
23. Partel V, Charan Kakarla S, Ampatzidis Y (2019) Development and evaluation of a low-cost and smart technology for precision weed management utilizing artificial intelligence. Comput Electron Agric 157:339–350. https://doi.org/10.1016/j.compag.2018.12.048
24. Ruigrok T, Henten E, Booij J, Boheemen K, Kootstra G (2020) Application-specific evaluation of a weed-detection algorithm for plant-specific spraying. Sensors. https://doi.org/10.3390/s20247262
25. Sanchez PR, Zhang H (2022) Evaluation of a cnn-based modular precision sprayer in broadcast-seeded field. Sensors 22(24):9723
26. Salazar-Gomez A, Darbyshire M, Gao J, Sklar EI, Parsons S (2022) Beyond map: towards practical object detection for weed spraying in precision agriculture. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9232–9238. IEEE
27. Zahidi UA, Cielniak G (2021) Active learning for crop-weed discrimination by image classification from convolutional neural network's feature pyramid levels. In: Vincze M, Patten T, Christensen HI, Nalpantidis L, Liu M (eds) Comput Vis Syst. Springer, Cham, pp 245–257
28. Grinsven MJJP, Ginneken B, Hoyng CB, Theelen T, Sánchez CI (2016) Fast convolutional neural network training using selective data sampling: application to hemorrhage detection in color fundus images. IEEE Trans Med Imaging 35(5):1273–1284. https://doi.org/10.1109/TMI.2016.2526689
29. Lai X, Liu J, Jiang L, Wang L, Zhao H, Liu S, Qi X, Jia J (2022) Stratified transformer for 3d point cloud segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8500–8509
30. Basha SHS, Pulabaigari V, Mukherjee S (2022) An information-rich sampling technique over spatio-temporal cnn for classification of human actions in videos. Multimed Tools Appl 81(28):40431–40449. https://doi.org/10.1007/s11042-022-12856-6
31. Leevy JL, Khoshgoftaar TM, Bauder RA, Seliya N (2018) A survey on addressing high-class imbalance in big data. J Big Data 5(1):1–30
32. Xu M, Yoon S, Fuentes A, Park DS (2023) A comprehensive survey of image augmentation techniques for deep learning. Pattern Recognit 137:109347
33. Chlap P, Min H, Vandenberg N, Dowling J, Holloway L, Haworth A (2021) A review of medical image data augmentation techniques for deep learning applications. J Med Imaging Radiat Oncol 65(5):545–563. https://doi.org/10.1111/1754-9485.13261
34. Khalifa NE, Loey M, Mirjalili S (2022) A comprehensive survey of recent trends in deep learning for digital images augmentation. Artif Intell Rev. https://doi.org/10.1007/s10462-021-10066-4
35. Jain S, Seth G, Paruthi A, Soni U, Kumar G (2022) Synthetic data augmentation for surface defect detection and classification using deep learning. J Intell Manuf 33(4):1007–1020. https://doi.org/10.1007/s10845-020-01710-
36. Akyon FC, Onur Altinuc S, Temizel A (2022) Slicing aided hyper inference and fine-tuning for small object detection. In: 2022 IEEE International Conference on Image Processing (ICIP), pp. 966–970. https://doi.org/10.1109/ICIP46576.2022.9897990
37. Rebuffi S-A, Gowal S, Calian DA, Stimberg F, Wiles O, Mann T (2021) Fixing Data Augmentation to Improve Adversarial Robustness
38. Chattopadhay A, Sarkar A, Howlader P, Balasubramanian VN (2018) Grad-cam++: generalized gradient-based visual explanations for deep convolutional networks. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 839–847. https://doi.org/10.1109/WACV.2018.00097
39. Petsiuk V, Das A, Saenko K (2018) RISE: Randomized Input Sampling for Explanation of Black-box Models
40. Huang G, Liu Z, Maaten L, Weinberger KQ (2018) Densely Connected Convolutional Networks
41. Tan M, Le QV (2019) Efficientnet: Rethinking model scaling for convolutional neural networks https://doi.org/10.48550/ARXIV.1905.11946
42. 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
43. Simonyan K, Zisserman A (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition
44. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. https://doi.org/10.1109/CVPR.2009.5206848
45. Rogers H, De La Iglesia B, Zebin T (2023) Evaluating the use of interpretable quantized convolutional neural networks for resource-constrained deployment

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.