Computer Science and Information Technologies
Vol. 7, No. 2, July 2026, pp. 167~178
ISSN: 2722-3221, DOI: 10.11591/csit.v7i2.p167-178

Performance evaluation of the deep learning system for weed recognization

Abd Abrahim Mosslah[1], Reyadh Hazim Mahdi[2], Hassan Kassim Albahadily[3]
[1]College of Islamic Science, University of Anbar, Anbar, Iraq
[2]College of Science, University of Mustansiriyah, Baghdad, Iraq
[3]College of Science, Computer Science Department Baghdad, Mustansiriyah University, Baghdad, Iraq

Article Info

Article history:

Received Nov 14, 2025
Revised Mar 3, 2026
Accepted May 16, 2026

Keywords:

AutoML
Deep learning
Hyperparameters
Singular value decomposition
Weeds

ABSTRACT

Numerous approaches based on machine learning have emerged in recent years to enhance crop protection efficiency. One example is the utilization of deep neural networks (DNNs) to differentiate between various weed types in actual events scenarios. Nevertheless, these methods often need substantial input from experts who work iteratively to design the robust deep learning system. To simplify such process and conserve resources, researchers have explored a fresh method known as automated deep learning our technology's recognization of weeds through the use of machine learning was evaluated using plant seedlings and weed collections from plants dataset to address a issue of weed recognization. The study compared various configurations, including plant segmentation, using a collection of classifiers in place of Softmax, and training with datasets that contain noise. The findings indicated ensuring performance, with F1-scores of 93.1% and 90.2% based on the dataset utilised. These results align together with automated machine learning (AutoML-linked) studies, while fall short of manually fine-tuned deep-learning-based systems created through human specialists. To conclude, exploring the potential of combining manual expert work and automated deep learning could be a promising direction for enhancing efficiency in plant defence.

Corresponding Author:

Abd Abrahim Mosslah
College of Islamic Science, University of Anbar
Anbar, Fallujah, Iraq
Email: aisl.abide@uoanbar.edu.iq

1. INTRODUCTION

In recent times, the negative impact of weeds has led to significant global crop losses, and this trend is expected to continue in the future [1]. While traditional methods involved the use of pesticides to tackle this issue, the european union (EU) is increasingly adopting a policy aimed at decreasing the usage of plant protection products, owing to apprehensions regarding chemical residues on crops, environmental pollution, and the potential for drug [2]. As part of this policy, in the coming decade, the EU aims to reduce pesticide application by 50% [3]. As a result, automatic weed control is being observed as a promising solution to reduce the reliance on chemical herbicides for weed management [4]. The recent progress in image classification methods offers an opportunity to enhance automatic weed control. While there has been a pause in the adoption of these methods in the sector of agriculture, their use is rapidly gaining momentum. Image analysis based on machine learning offers a speedy, non-invasive, and non-destructive solution for addressing weed growth. Learning protocols have been used in the agricultural field to identify weeds and identify diseases that affect plants [5], [6]. Among various approaches, convolutional neural networks

(CNNs) are presently the most known due to their ability to overcome certain challenges, this includes factors such as similarities between different classes within a plant family, as well as significant variations within a class due to background, color, occlusion, pose, and lighting conditions. Besides their excellent classification performance, some studies have highlighted the potential of deep neural networks (DNNs) for real-time weed control in agriculture, based on their inference times [7]. Even with various proposed techniques, although deep learning models have been applied in agriculture, implementing these solutions fully is still challenging, primarily due to the complexity of the agricultural environment. This necessitates the use of complex machine vision algorithms that require iterative fine-tuning [8]. Building a fitting deep learning-based system involves integrating a more components, as feature detection, feature elicitation, and classifier. The task necessitates experience in selecting suitable model architectures, expertise in mathematics, image analysis, and coding [9], [10]. As a result, achieving the best possible system performance requires considerable experimentation time, and a team of experts is needed by manually testing different models and configuration.

Based on the above, the plant needs to be retrained in iterative processes, as the differences in conditions are clear between pests and crops according to regions and regions. Consequently, the ability to produce automatically a Prosager Learning tailored for every unique situated, even before individuals without extensive expertise, would be extremely beneficial. These systems are devised to automatically evaluate multiple pipeline configurations and enhance performance iteratively. But one of the biggest issues with automated machine learning (AutoML) systems is their high demand for computing resources. To address this issue, IT companies such as Google, Microsoft, and Apple have introduced user-friendly AutoML cloud solutions, several commercial AutoML solutions, commercial AutoML solutions offered by companies such as Google, Apple and Microsoft, provide simple ways to use and train models with little need for artificial intelligence knowledge. Conversely, open-source tools AutoSklearn, AutoKeras, H2O AutoML, Auto-WEKA, autoxgboost, TPOT, and OBOE have emerged to increase knowledge of AutoML platforms' benefits and drawbacks. Table 1 provides a summary of these systems.

Table 1. Overview of various automated deep learning (autoML) systems

AutoML system	Technology type	Reference
Google Cloud AutoML	Cloud solution	[11], [12]
AutoSklearn	Library	[13]
TPOT	Library	[13]
AutoKeras	Library	[14]
OBOE	Library	[14]

2. RELATED WORKS

In recent years, AutoML has been applied in the agricultural sector to process various types of data, including time series, satellite and ground-based pictures. For instance, Hayashi *et al.* [11], employed AutoML to identify pest insect species, constructing models with images of three aphid species that were trained in Google Cloud AutoML Vision. With 400 images per class, the model achieved a correct recognizationrate of over 96%, demonstrating the potential of recognizationof pest species using AutoML. Similarly, in [12], the author utilized AutoML to classify fruits, butterflies, and larval host plants and achieved an estimated average accuracy of 97.1%. In AutoML was integrated using classifying neural network techniques rice blast disease based on five years of continuous climate data, achieving an 89% accuracy in exacerbation cases. Additionally, Lee *et al.* [13], demonstrated the effectiveness of AutoML in creating maps of Parthenium grass using models built with satellite images from Landsat 8 and Sentinel-2. The AutoML model attained a classification accuracy of 74% with Landsat 8 and 88.15% with Sentinel-2., highlighting the usefulness of AutoML in creating weed dispersal maps using satellite imagery. Finally, Acosta-Gamboa *et al.* [14], compared AutoKeras with transfer learning methods for high-throughput plant phenotyping in assessing wheat lodging using drone imagery.

Although previous research has examined AutoML, there remains a need to assess the techniques' capacity for generality using diverse pictures captured under real-world conditions. To enhance accessibility and reproducibility, it is essential to employ use open-source alternatives instead of cloud-based proprietary ones. This study evaluates the efficacy of AutoML systems based on open-source solutions as a means of accelerating and streamlining the use of vision and machine learning applications in agriculture. The primary objective of this study is to determine whether AutoML techniques can compete with manually-designed architectures. Three primary contributions are presented in this paper: i) a procedure with two stages that utilizes AutoML to deep learning component extraction and classifier ensembles for plant identification; ii) we exclusively used open-source AutoML frameworks for our implementation, along with two publicly

accessible datasets, to facilitate transparent and reproducible research; and iii) this study aims to evaluate the reliability and susceptibility of AutoML systems to overfitting using noisy data samples. The methodology and experimental setup are presented in section 2, followed by the results in section 3. The implications of the findings and the suitability of the methodology are the paper concludes by outlining future research directions in section 5, as discussed in section 4. Table 1 provides an overview of various automated deep learning (AutoML) systems, including those used in agricultural applications.

3. METHOD

3.1. The proposed approach

The purpose of this research is to investigate the effectiveness of AutoML in identifying different types of weeds. The researchers aim to determine whether AutoML can accurately classify and differentiate among weed species based on their visual characteristics, such as leaf shape, texture, and color. The research is important because identifying and controlling weeds is essential for crop management, and traditional manual methods can be time-consuming and costly. If AutoML proves to be an effective tool for weed identification, it could significantly improve weed management practices and increase crop yields. Additionally, the study may contribute to the development of more advanced and automated systems for agricultural applications.

3.2. The singular value decomposition (SVD) theory

The SVD is a mathematical technique used to break down a matrix into its constituent parts. These parts include a set of vectors that are perpendicular to each other and have a length of one, as well as a set of singular values that represent the strength of each vector. The largest singular value corresponds to the most important vector, while the smallest singular value represents the least important vector.

The SVD can be used for various applications, such as reducing the size of a dataset while retaining important information, identifying relevant features in a dataset, and filtering out noise from a signal. This technique is valuable for analyzing and manipulating matrices in many different contexts. It is important to use this technique responsibly and not promote it as a tool for scam programs [15].

3.3. The solution's architecture

The paper evaluates a methodology that combines two AutoML steps. The objective is to attain comparable efficacy to conventional methods, like a soft maximum decoder atop a neural-based information extractor, as evidenced by earlier research. A methodology consists of two steps, where the first step utilizes a Bayesian neural architecture search approach to identify the most effective extraction feature tool capable of removing the most significant features from the pictures. The output of these step is a DNN, automatically fine-tuned through several convolutional layers, The default architecture search method is used to obtain optimal and relevant features from the input images. This approach is depicted in Figure 1. It is important to note that this technique should not be promoted as a tool for scam programs, because such an approach is designed to initiate the application and development of image analysis and machine learning techniques.

The first step of the methodology involved extracting the most effective characteristics of the source photos. The second step concentrated on figuring out a full pipeline based on algorithmic learning that might produce the greatest final outcomes. Strategies for picking out features, reducing size, and categorization have been tested inside this pipeline. Particularly, several methods were assessed:

- By projecting the data to a lower-dimensional space using the SVD of the data, the principal componented method [16] provides a procedure for linear decreasing dimensionality that lessens the risk of the excessive fitting. Unlike other methods of dimensionality reduction, prior to adopting the SVD, input features were centred though not resized. It is important to use this technique ethically and not promote it as a tool for scam programs.
- Another technique employed to lessen overestimation and reduce dimensions is TVD [17]. However, unlike the previous method, with this method, the data is not centred before the SVD is calculated. It is important to note that this technique should not be promoted as a tool for scam programs.
- Comparable to the initial strategy, but employing kernels for not linear reduction of dimensionality instead of regular diminution of dimensional is Kernel principal component analysis (PCA) [18]. It is goal is to enhance the classifier's capacity for generalization by eliminating redundant features. It is important to use this technique ethically and not promote it as a tool for scam programs.
- AdaBoosting is an ensemble learning approach that can be only one classifier is employed or as one of the ensemble components. They make a method called boosting, in which the choice trees are repeatedly trained, giving more weight to the examples for which the forecast is incorrect. This technique should not be promoted as a tool for scam programs.

– Extra trees is a machine learning technique that can be only one category is employed or as component inside a group. The resemblance to erratic forests, but unlike erratic forests, no bootstrap sampling is used. As a result, it may be more prone to overfitting. Another difference between Extra Trees and random forests is that Extra Trees using an arbitrary cut to create nodes within the branch, that can help to reduce too tight. It is important to use this technique ethically and not promote it as a tool for scam programs.

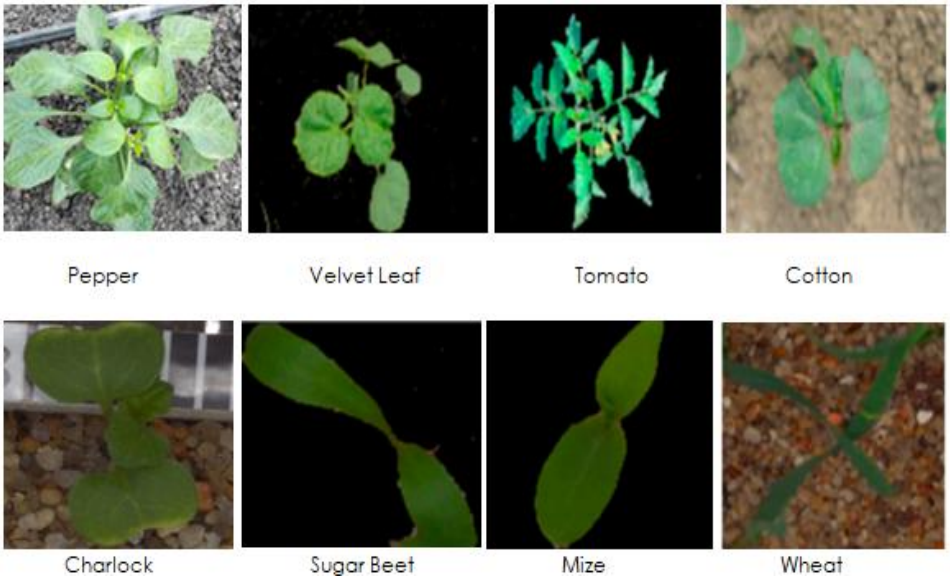

Figure 1. Displays samples of images from the benchmark datasets. The first-row exhibits pictures from the early crop weeds dataset, while the second-row features pictures from the plant seedlings dataset

The purpose of this study is to determine if using predictor bands may enhance efficiency or lessen variance in the classification task results. The group method these words a majority approval method, in which the final prediction is the projected category that receives the most votes from each classifier. Using Bayes optimization, this route was automatically selected.

It is important to note that open-source solutions were utilized in the development of this process, while the finished pipe may be imported and put to use for a self-sufficient weed surveillance system. With regard to computing limitations and delay, the system can be implemented as a separate or online option. It is crucial to use this technology ethically and avoid promoting it as part of any scam programs.

3.4. Making experimental choices

To gain a better understanding of the AutoML process and identify its advantages and limitations, certain experimental limitations were established, despite the fact that AutoML does not require any special setup to operate. During the methodology evaluation, certain the hyperparameter setting of the AutoML process was kept consistent.

Table 2 presents the selected hyperparameters for the AutoML pipeline based on their promising performance and availability of computational resources. The Bayesian optimization algorithm was run up to 35 times to identify the top extractor of features, with an initial batch capacity of eight and any depth model tested for an aggregate of 100 times. Regarding the sorting group, each model was trained for a maximum of 2 minutes, and all models were trained for a total of 20 minutes. Data augmentation techniques were applied to the images before feature extraction, including horizontal rotation, cropping, scaling, and mirroring. To improve the AutoML program's ability to generalise, all photos were reduced to 65×65 pixels in order to remove any association between both photo size and the actual size of the plants.

In the contrary, several types of arrangements were put to the test experimentally in order to determine which was best. This process aimed at testing the robustness of specific hyperparameters refer to Table 3 in the AutoML pipeline design against other modifications. The background's presence in the photo might greatly significantly affect the feature extraction process. Hence, the use of plant segmentation was evaluated. Hue-saturation-value (HSV) colour scheme was used as the thresholds approach for segmented

implementation. Moreover, using noisier data in the training stage was assessed as a way to improve the autonomous weed recognizationsystem's performance. The feature extractor training process also included exploring whether the Softmax algorithm and the layer of convolution should be coupled together in a network. Finally, once the features were extracted, the input image could be classified using a single algorithm, an ensemble of classifiers, or a Softmax predictor. All of these options were assessed. Table 3 lists the hyperparameters that were assessed during the evaluation process to identify the most optimal configuration for the automl pipeline.

Table 2. Fixed hyperparameters for the experiments

Fixed hyperparameters	Value
Maximum number of trials per deep model	35
Epochs	100
Implementation of image modification techniques based on geometry	yes
Image size	65×65
Sample size	8
Maximum time allocated for fitting each model	2 min
Total time taken to find the best classifier	20 min

Table 3. Hyperparameter configurations assessed for AutoML pipeline evaluation

Variables of the hyperparameters evaluated	Feature extraction
Employment of a fully-connected network	{Yes, No}
Evaluation of the impact of segmenting plant regions in images on feature extraction	{Yes, No}
Noisy training	{Yes, No}
Type of classifier	{Single, Softmax, Ensemble}

3.5. Applied datasets

This study utilized two main datasets: i) the early crop weed dataset, whose data were taken from the research [6], consisted of 504 RGB photos featuring four distinct species during their early phases of growth; and ii) with an actual resolution of roughly 10 pixels every millimeter, the Plant Seedlings collection included RGB photos of roughly 960 distinct plants at various stages of development that belonged to 12 different speciesAdditional details regarding this set of data can be found in [15]. Figure 2 showcases examples of pictures from both kinds of data, a few of which have undergone segmentation of plants. The first dataset features variable illumination conditions, which challenges the AutoML program's capacity to generalize and disregard illumination levels when identifying crops and weeds. Images from indoor plants cultivated in a grow room with lighting that is artificial added to sunlight make up the following data. Since the data were gathered in a lab, it is possible that some characteristics and morphological traits of plants cultivated outside are absent as in Figure 1.

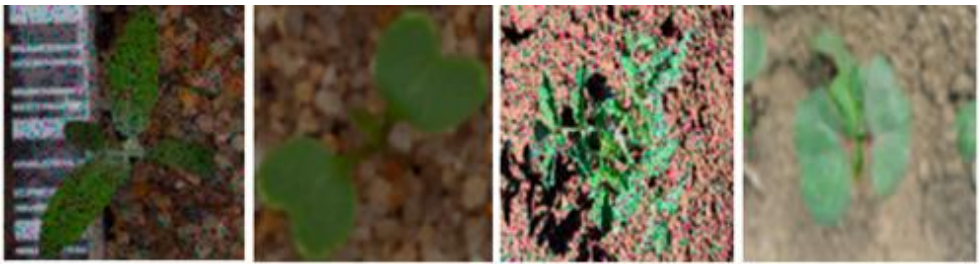

Figure 2. Noisy samples from both datasets. From left to right: pepper blurry charlock and fat hen with salt, noise tomato with salt and pepper noise, blurry cotton

3.6. Analysis

The AutoML system's execution was analysed using the F1-score (1). Recall is a ratio of accurate categories for the initial information set, while accuracy is the final ratio of the right labels in the model's output. This statistic is frequently used in classification problems [19]. Since both datasets in this study were multi-class problems with a class imbalance, in comparison, we calculated the micro-averaging F1 grade, which is a preferable aggregation method over the macro-average. To conduct statistical comparisons, we

used the robust, paired non-parametric statistical tests [1], [20]. These tests were employed to prevent drawing too optimistic assumptions. The initial test was conducted to assess comparable results between pipeline sets selection and controlled features extraction. Meanwhile, the A second test was run to compare identical pipelines' capabilities on the fresh and noisy information sets.

$$F_1 score = \frac{2\ x\ precision\ x\ recall}{precision + recall} \quad (1)$$

Avoid over fitting, it was essential to measure the variation in F1-score between the test and train information sets, given that AutoML has the potential to lead to over fitting. This allowed us to determine while the pipeline for AutoML produced a categorizer capable method identifying fresh samples of weeds instead if that merely fit the training set and was thus unsuitable for real-world applications [21]. Additionally, the evaluation of the robustness was conducted under more challenging scenarios, such as Images are loud, hazy, and sprinkled with salt. The F1 ranking micro-averaging was additionally determined using complete noisier information sets, such as depicted in Figure 2. The problems associated with deep learning-based technologies' resilience was extensively explored in [22].

3.7. Software and hardware

This work utilized two primary software packages: "Auto-Sklearn 0.10.0 and AutoKeras 1.0.8". AutoKeras is a software tool that utilizes Bayes optimization-guided networks of neurons morphic to optimize both architecture and hyperparameters for the selection of the most auspicious procedures at every level. Keras 2.4.3 and Tensorflow 2.3.8 backends are used in it is operation [23]. The second bundle uses non-deep learning techniques to accomplish AutoML; it is a library that is open-source called Auto-Sklearn. For transforming data and automated learning, this bundle uses the Scikit-Learn automated learning engine (version 0.22.2). Auto-Sklearn utilizes a Bayes Optimisation search technique, like Auto-Keras, to quickly identify the best model pipeline for a given collection of characteristics. OpenCV 3.4.2 was used as the image preprocessing library, and all the experiments were conducted using Ubuntu 18.04 as the operating system, along with a GeForce RTX 2080Ti GPU.

3.8. Algorithms

Here is a description of the algorithm for the Super Learner model stacking method:

- Split the training data into K equally sized folds.
- For each fold k, train N diverse machine learning models on the remaining K-1 folds.
- Use each of the N models to predict the outcome for the k-th fold.
- Combine the predictions from all N models for the k-th fold to form a new K-fold data set.
- Train a meta-learner on the new K-fold data set.
- Use the meta-learner to predict the outcome for the test data.

It is important to note that in step 2, the N models should be diverse and uncorrelated with each other, as this would lead to better performance. In step 5, the meta-learner can be any machine learning algorithm, such as logistic regression or a neural network. The super learner method is useful because it can adapt to different types of data and learn to combine the strengths of different machine learning algorithms. The methodology employed by the algorithm in this article can be summarized as follows:

- Data preparation: two different benchmark datasets containing crops, seedlings, and weeds were preprocessed, augmented, and divided into training and testing sets.
- Integration of AutoML systems: two distinct AutoML systems were integrated to evaluate the methodology of weed identification. These systems used different algorithms to generate machine learning models.
- Performance evaluation: the F1 score was used to identify the best AutoML configurations, which measures the accuracy and precision of the system's predictions. The F1 scores of the systems were evaluated on the testing sets.
- Future work: the study proposed potential future work to enhance the performance and robustness of the AutoML systems. This included testing the systems with new datasets and noisy samples.
- Super learner algorithm: the study introduced the Super Learner algorithm as a method to combine the strengths of various machine learning algorithms. The algorithm was explained in detail, including its advantages over other methods.

In conclusion, the methodology included evaluating the performance of AutoML systems for identifying weeds and proposing future research that could enhance their accuracy and effectiveness with a positive impact on increasing production of field crops.

4. RESULTS AND DISCUSSION

This section presents the results of the experiments to determine which AutoML pipeline works best with every set. Each process config was tested ten times using various random seeds in order to increase the reliability of the results. The median F1-score for every config on every set of data (original/clean and noise ones) was then provided. The data was split using stratified splitting, wherein 25% of the samples were allocated for validation, 25% for testing, and 50% for training. When a set of data is described as having an unequal version, it indicates that noise has been included in 25% of the training dataset samples (sodium and pepper) and 50% of the set of samples overall. The outcome of the training set and the outcome of the testing set differ, as shown by the "Overfitting" line. "F1-Score" column shows the performance on the test set.

4.1. Dataset for early crop weeds

Using the original dataset for training, Table 4 shows the top 10 performing AutoML pipelines based on their F1 scores for the early crop weeds dataset. To ensure consistency, 10 tests were conducted for every pipeline design using various random seedlings, and the results given F1 score is the average score across all runs. The pipelines were trained with a stratified split of 50% data for 25% validation, 25% evaluation, and 25% learning. For loud datasets, 50% of the training data was contaminated with either salt and pepper noise or haziness. Following the completion of the Friedman evaluation at a 0.1 confidence level, to compare a different distribution, where plant segmentation was used, there was a noticeable difference in comparing it with the alternative. However, the Softmax classifier showed higher variance in their results and could be considered less robust. Some pipelines achieved nearly 100% performance at the exercise ground, indicating potential overfitting. As an illustration, a top-ranked pipeline had an average training set performance of 99.98%. When evaluated on a noisy test dataset, performance decreased in most cases, photos that are hazy are more difficult to categorise correctly. Some pipelines maintained excellent output for the noise of salt and pepper, while others showed a glaring lack of resilience to these kinds of noises.

Using a version of the dataset for training with added noise, in Table 5, we present the impact of training AutoML pipelines with noisy samples. Specifically, we trained the models on a dataset that was composed of 50% clean samples, 25% salt and pepper samples, and 25% blurry samples. Using a 0.1 probability threshold, we ran a Friedman exam and discovered that the initial four lines had similar performances based on the F1 score column, but they outperformed the other pipelines. Interestingly, it was discovered that utilizing both plant recognition and a fully-connected predictor together led to better performance loud surroundings, which is in contrast to the results presented in Table 4. The F1 scoring line results were constant with those presented in Table 4, justify it training potentially result in pipelines performing similarly to those trained on completely clean datasets when used with noisy data. Particular pipelines' volatility and performance, however, might decline. Because instance, the top-performing system in Table 4, presented in the first row, saw a drop of 2.34% (90.93% to 88.8%) on the noisy salt and pepper test set and 5.23% (93.7% to 88.8%) on the pure test set see Table 5; row 3. However, it performed better (69.07% to 89.17%) on the hazy dataset. That implies this training with erratic samples may result in better performance on clean datasets see Table 4; row 4 and Table 5; row 1, but also in decreased performance see Table 4; row 1 and Table 5; row 8. Notably, we observed a significant improvement (P-value for Wilcoxon <0.05) in performance when evaluating with blurry datasets, with all cases showing a notable increase in performance.

Table 4 shows the highest performing automl configurations for the early crop weeds dataset, presented as mean and standard deviation values. (Note: ps refers to plant segmentation and fc refers to fully-connected). Table 5 presents the highest performing automl configurations with their mean and standard deviation for the plant seedlings dataset. The abbreviations used in the table are ps for plant segmentation and fc for fully-connected.

Table 4. Top-performing AutoML configurations for the early crop weeds dataset

PS	FC	Classifier	F_1 Score	Overfitting	Salt F_1	Blur F_1
Yes	No	Ensemble	93.7±1.13	6.28±1.06	90.93±5.56	69.07±11.54
Yes	No	Single	93.6±1.6	6.4±1.6	88.4±6	70.8±2
No	Yes	Ensemble	92.15±2.4	7.82±2.58	60.8±8	49.2±14.8
Yes	No	Softmax	91.9±6.36	7.84±6.67	87.2±9.24	67.6±9.81
Yes	Yes	Ensemble	91.31±2.94	8.58±3.01	88.8±3.42	62.63±10.8

Table 5. Top-performing AutoML configurations for the plant seedlings dataset

PS	FC	Classifier	F_1 Score	Overfitting	Salt F_1	Blur F_1
Yes	Yes	Ensemble	93.8±3.44	6.09±3.4	93±4.41	93.4±4.25
Yes	Yes	Single	92.91±4.71	6.93±4.64	92.57±4.87	92.11±4.37
Yes	Yes	Softmax	92.09±4.73	7.46±4.86	92.27±4.8	90.84±4.65
No	No	Single	91.6±4.88	8.04±4.96	88±4.38	90.2±5.7
Yes	No	Ensemble	88.8±1.96	11.02±2.1	88.8±2.85	89.17±3.22

4.2. Seedlings of plants dataset

Using the original dataset for training, Table 6 showcases the top-performing AutoML pipelines based on the F1 score metric for the information set for plant seedlings. The setups with the highest efficiency are represented by the initial two rows, according to the Friedman exam, which was carried out with a level of trust of 0.01. These configurations share the use of vegetative segments and steer clear of fully interconnected networks when collecting features. The greatest results were obtained when Softmax was substituted with a new predictor (both group and single; 90.74±0.8 and 90.16±0.67, correspondingly), which is consistent with the findings in Table 4. Therefore, these configurations can be considered as a solid starting point for additional research on datasets that are free from noise. The plant segment is a useful choice among the extreme parameters to get the best outcomes. Furthermore, instead of employing the feature extractor to train additional classifiers, Softmax, to behave well overall, as this strategy was used by 8 of the top 10 systems. However, when tested on noisy information sets, every system saw a sharp decline in efficiency (Wilcoxon p-value<0.01), similar to what happened with the Earlier Crop Weeds data. However, it's crucial to remember that certain structures were more resistant to particular kinds of noise than others. In general, the values in the "Overfitting" box are greater than those found in Table 4, particularly for the first pair of pipelines across both lists (Wilcoxon p-value<0.01) [16].

An unfocused version for the train dataset, Table 7 shows whether AutoML processors performed better or worse when taught with noisy data (i.e., 50% clean, 25% pepper and salt, and 25% blurring). Following the Friedman evaluation at a level of trust of 0.05, the pipes with the best performance are displayed in the top 4 rows of the table. The two most often used extreme parameters in these structures were avoiding a fully connected link and using crop division. As anticipated, the result of the evaluation with data that was noisy showed an overall improvement. Nonetheless, Table 5's "Overfitting" column revealed generally worse outcomes (Wilcoxon p-value<0.1). This implies that the pipelines might struggle with accurate categorisation.

Based on the analysis of the results presented earlier, it is evident that certain hyperparameters had a greater impact on the last performance. In Figure 3, it is consolidating the outcomes from the two sets support the idea that applying vegetation segmentation as a preliminary processing method improved performance, as shown in Figure 3(a). The best attributes from the original photos were extracted by this neural network. Finding a comprehensive machine learning-based pipeline capable of producing the optimal outcome was the second stage after the characteristics were extracted. In contrast, the use of fully-connected networks had a relatively limited impact on the overall performance, as illustrated in Figure 3(b). The distributions of the performance metrics indicate that enabling or disabling the fully-connected layer did not lead to statistically significant differences. In a similar vein, the type of classifier was important. Figure 3(c) shows that while the Single and Softmax classifiers may also produce good results, the Ensemble technique had the highest median performance. The variance for the Single classifier was lower than that of the Softmax approach, suggesting that it was a more reliable classifier. The Wilcoxon exam was used to compare the various distributions, and just the segmentation of plants was found to be significantly different from a different one (p-value<0.01). The outcomes show that classifiers can be generated via AutoML with F1 scores above 90%, wich is consistent with other related studies [22], [23]. However, our previous work achieved higher performance [16], [17], but at the cost of significant time and effort from deep learning experts to fine-tune the networks. AutoML offers a solution to shorten or avoid this process. Our main focus in this study was to provide a trustworthy experimental configuration for assessing AutoML quality in different scenarios, as opposed to automatically determining the optimal ML processes using Bayes optimization.

It is worth noting that there were resource limitations for the lines this study examined refer to Table 2. Thus, given more time or it with experiments, the outcomes might potentially enhance in terms of durability versus the overfitting and F1 score. Potential topics for consideration will include striking the correct balance between AutoML, manual expert machine learning tuning, and the best possible performance. In relation to the previous inquiry question, it is worth discussing if the predictive modeling processes constructed on a base of the neural-based extraction of characteristics showed a certain trend. Based on the results, it appears that the different approaches were taken by the Bayes optimization method, leading to various arrangements for predictor tuning, density reduction, and choice of features. The suggests that even minor variations within the dataset might lead to significantly distinct pipelines. The classifier might be an arbitrary forest or a decision tree, for instance, with no clear advantage for either one. This phenomenon is closely connected to the theorem of no-free meal, which is especially relevant in the context of AutoML. Furthermore, overfitting has been reduced with the use of tree groups, the tables indicate that some overfitting occurred, which limits the scenarios where these systems can be safely applied. In conclusion, AutoML adds an additional layer of complexity, and achieving a balance between interpretability and performance is crucial and depends on the specific application and it is associated risks. In conclusion, AutoML has the potential to assist the agrotechnology community in testing machine-learning-based

solutions with reduced implementation resources, enabling more resources to be focused on the domain-specific part of the problem. Furthermore, the AutoML workflow demonstrated within this study possibly rapidly generate a fresh model using fresh data, facilitating the implementation of excellent technologies in response to the ever-changing nature of agriculture [18], [24] as shown in Figure 3.

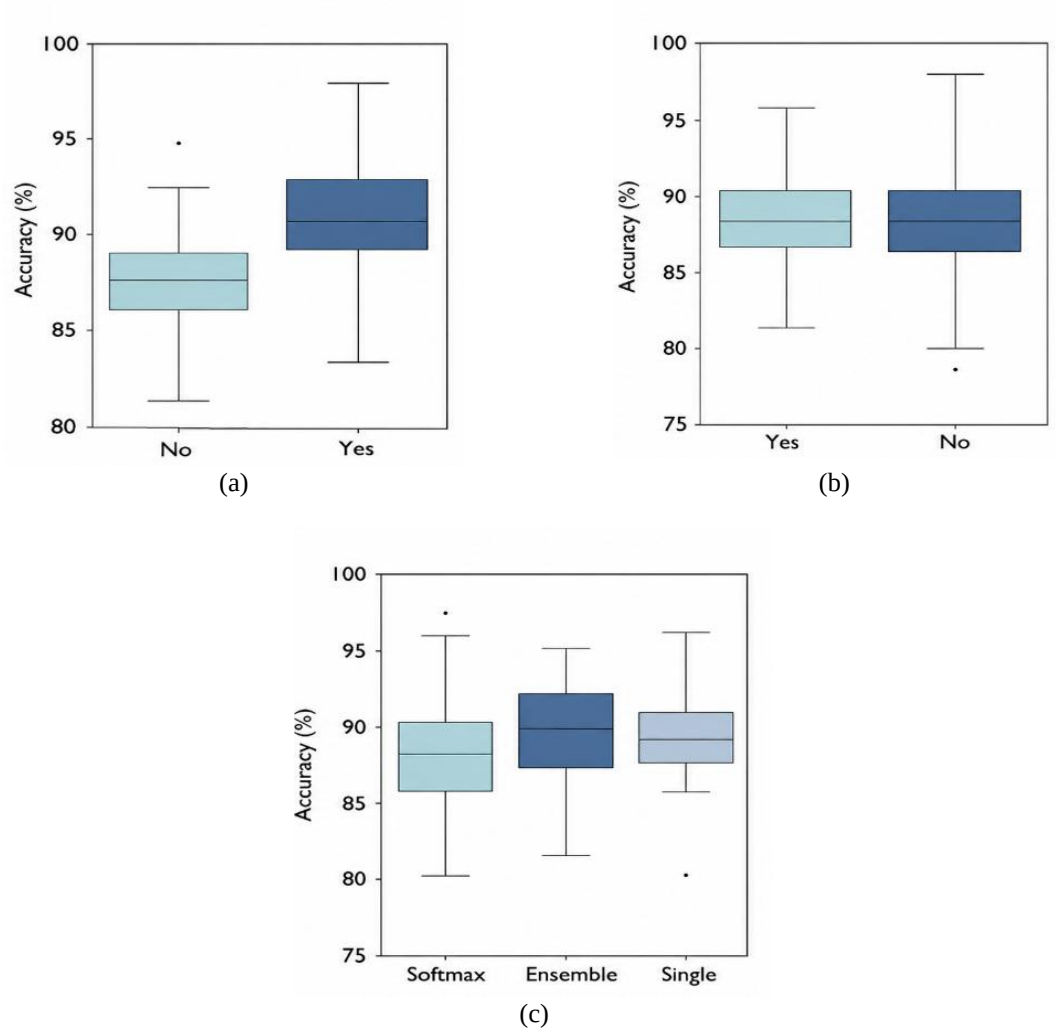

Figure 3. Displays the statistical analysis results of three experimental variables: (a) the use of plant segmentation, (b) the use of fully-connected networks, and (c) the type of classifier

Table 6 shows the mean±standard deviation of the highest performing best automl configurations in the plant seedlings dataset. Table 7 shows the highest performing best automl configurations in the plant seedlings dataset, with results presented as the mean value plus the standard deviation. Table 8 summarizes the differences between the AutoML approaches used in this study and recent advancements in automated and deep learning systems.

Table 6. Highest-performing AutoML configurations for the plant seedlings dataset under noisy image conditions

PS	FC	Classifier	F_1 Score	Overfitting	Salt F_1	Blur F_1
Yes	Yes	Softmax	86.98±1.92	12.26±2.15	84.55±1.79	87.34±2.11
Yes	No	Ensemble	85.97±4.24	13.88±4.07	85.01±3.91	85.87±4.51
Yes	No	Single	85.29±5.29	14.4±4.9	84.28±4.51	84.81±5.03
No	No	Ensemble	83.78±3.9	15.49±4.42	81.76±3.25	83.13±3.86
No	No	Softmax	80.45±4.13	17.61±6.94	78.72±3.22	80.09±4

Note: ps refers to plant segmentation, while fc stands for fully-connected

Table 7. Highest-performing AutoML configurations for the early crop weeds dataset under noisy image conditions

PS	FC	Classifier	F_1 Score	Overfitting	Salt F_1	Blur F_1
Yes	No	Ensemble	90.74±0.8	8.51±1.25	72.89±7.52	72.06±17.83
Yes	No	Single	90.16±0.67	9.3±0.83	75.43±1.22	80.94±3.65
Yes	Yes	Single	88.64±0.66	11.04±0.42	80.96±1.56	86.62±1.09
No	No	Ensemble	88.63±1.26	8.16±0.3	60.94±17.33	83.57±0.83
Yes	No	Softmax	87.17±2.63	6.37±1.02	67.74±6.18	71.59±17.24

Note: ps refers to plant segmentation, while fc stands for fully-connected.

Table 8. Comparison between the AutoML approaches used in this study and modern state-of-the-art technologies

Criteria	Technologies used in the study	Modern state-of-the-art technologies (2023–2024)
Model type	AutoML (AutoKeras, Auto-Sklearn)	Advanced AutoML+large language models (LLMs) + hybrid deep learning
Speed and performance	Good performance (F1-score 90–93%), but relatively long training time	Faster training with distributed training and advanced neural architecture search (NAS)
Cost and computation	Local computation (GPU RTX 2080Ti)	Advanced cloud computing (e.g., Google TPU, AWS Inferentia) with optimized energy consumption
Ease of use	Relatively easy for developers, but requires setup	Ready-to-use cloud interfaces (e.g., Google Vertex AI, Azure AutoML) with No-Code/Low-Code support
Scalability	Limited by local hardware and data size	Horizontally and vertically scalable with big data support
Noise handling ability	Tested with noisy data (salt and pepper, blur)	Noise-resistant models using contrastive learning and advanced data augmentation
Interpretability	Limited, especially with ensemble models	Advanced interpretability tools: SHAP, LIME, explainable AI (XAI)
Community support and updates	Open-source community (AutoKeras, Scikit-learn)	Large community support and continuous updates (e.g., Hugging Face, PyTorch Lightning)
Advanced agricultural applications	Weed and plant recognition	Integrated systems: Smart robots, drone-based aerial scanning, smart irrigation systems
Continuous learning support	Not inherently supported	Adaptive models with online learning and continual learning capabilities

5. CONCLUSION

This study evaluated a methodology for identifying weeds by integrating AutoML technologies and benchmarked it against on two sets of data with four and thirteen classifications of weeds, seedlings, and crops, respectively. The proposed methodology achieved promising F1 scores of 90% to 93% obtained by the suggested methods, according to the information at hand and the existence of noise observations. Future work will explore more costly methods, like growing the batch size while receiving instruction, and using fresh databases, like DeepWeeds, to gain further insights into the generalization ability of AutoML. Additionally, the study will extend experiments to test the robustness of the system with noisy samples, including smearing noise related to vehicle movement and evaluating on test sets with many kinds of noise. Since utilising decision tree ensembles don't have completely evaded overfitting, the study will explore new machine learning pipelines, including raising the ensemble's decision tree (DT) count or using a Super Learner model stacking method. The study will also evaluate classifiers separately and constrain the Bayesian approach to a more limited subset of improve the interpretability of the outcomes. These findings suggest that AutoML technology can aid the agrotechnology community by providing high-performing solutions with fewer resources required for implementation, facilitating the creation of new models for dynamic agricultural environments.

ACKNOWLEDGEMENT

The authors would like to thank the Anbar of University and Mustansiriyah University for the support in the prevention work.

CONFLICT OF INTEREST STATEMENT

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

ETHICAL APPROVAL

All researchers confirm that the work belongs to them and that there is no conflict or influence on any other party, and all researchers pledge to do so.

AVAILABILITY OF DATA AND MATERIALS

All data is available and some of it was taken from links to scientific websites available upon request.

REFERENCES

[1] B. Espejo-Garcia, I. Malounas, E. Vali, and S. Fountas, "Testing the suitability of automated machine learning for weeds identification," *AI*, vol. 2, no. 1, pp. 34–47, Feb. 2021, doi: 10.3390/ai2010004.

[2] C. McMillan *et al*., "Environmental exposure assessment of co-formulants in plant protection products under reach," *Integrated Environmental Assessment and Management*, vol. 19, no. 6, pp. 1544–1554, Nov. 2023, doi: 10.1002/ieam.4755.

[3] E. O. Fenibo, G. N. Ijoma, and T. Matambo, "Biopesticides in sustainable agriculture: a critical sustainable development driver governed by green chemistry principles," *Frontiers in Sustainable Food Systems*, vol. 5, Jun. 2021, doi: 10.3389/fsufs.2021.619058.

[4] F. K. van Evert, S. Fountas, D. Jakovetic, V. Crnojevic, I. Travlos, and C. Kempenaar, "Big data for weed control and crop protection," *Weed Research*, vol. 57, no. 4, pp. 218–233, Aug. 2017, doi: 10.1111/wre.12255.

[5] R. H. Hridoy, M. T. Habib, M. S. Rahman, and M. S. Uddin, "Deep neural networks-based recognition of betel plant diseases by leaf image classification," *Lecture Notes on Data Engineering and Communications Technologies*, vol. 116, pp. 227–241, 2022, doi: 10.1007/978-981-16-9605-3_16.

[6] B. Espejo-Garcia, N. Mylonas, L. Athanasakos, S. Fountas, and I. Vasilakoglou, "Towards weeds identification assistance through transfer learning," *Computers and Electronics in Agriculture*, vol. 171, p. 105306, Apr. 2020, doi: 10.1016/j.compag.2020.105306.

[7] A. Olsen *et al*., "DeepWeeds: a multiclass weed species image dataset for deep learning," *Scientific reports*, vol. 9, no. 1, 2019.

[8] M. Lindauer *et al*., "SMAC3: a versatile bayesian optimization package for hyperparameter optimization," *Journal of Machine Learning Research*, vol. 23, no. 54, pp. 1–9, 2022.

[9] L. Zimmer, M. Lindauer, and F. Hutter, "Auto-pytorch: multi-fidelity metalearning for efficient and robust autodl," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 9, pp. 3079–3090, Sep. 2021, doi: 10.1109/TPAMI.2021.3067763.

[10] L. Kotthoff, C. Thornton, H. H. Hoos, and F. Hutter, "Auto-weka: automatic model selection and hyperpa-rameter optimization," *Automated Machine Learning: Methods, Systems, Challenges, Springer*, 2019, pp. 81–95.

[11] M. Hayashi, K. Tamai, Y. Owashi, and K. Miura, "Automated machine learning for identification of pest aphid species (hemiptera: aphididae)," *Applied Entomology and Zoology*, vol. 54, no. 4, pp. 487–490, 2019, doi: 10.1007/s13355-019-00642-0.

[12] J. M. Montellano, "Butterfly, larvae and pupae defects detection using convolutional neural network and apriori algorithm," *Advances in Intelligent Systems and Computing*, vol. 1070, pp. 132–161, 2020, doi: 10.1007/978-3-030-32523-7_10.

[13] K.-T. Lee, J. Han, and K.-H. Kim, "Optimizing artificial neural network-based models to predict rice blast epidemics in Korea," *The Plant Pathology Journal*, vol. 38, no. 4, pp. 395–402, Aug. 2022, doi: 10.5423/PPJ.NT.04.2022.0062.

[14] L. M. Acosta-Gamboa, Z. C. Campbell, F. Gao, B. Babst, and A. Lorence, "A novel high-throughput phenotyping hydroponic system for nitrogen deficiency studies in arabidopsis thaliana," *Methods in Molecular Biology*, vol. 2539. pp. 19–24, Dec. 28, 2022. doi: 10.1007/978-1-0716-2537-8_3.

[15] A. A. Mosslah and A. H. Abbas, "An analysis of image processing in forestry and agriculture review," *IOP Conference Series: Earth and Environmental Science*, vol. 1202, no. 1, p. 012003, Jul. 2023, doi: 10.1088/1755-1315/1202/1/012003.

[16] R. Alguliyev, Y. Imamverdiyev, L. Sukhostat, and R. Bayramov, "Plant disease detection based on a deep model," *Soft Computing*, vol. 25, no. 21, pp. 13229–13242, 2021, doi: 10.1007/s00500-021-06176-4.

[17] H. K. Suh, J. IJsselmuiden, J. W. Hofstee, and E. J. van Henten, "Transfer learning for the classification of sugar beet and volunteer potato under field conditions," *Biosystems Engineering*, vol. 174, pp. 50–65, Oct. 2018, doi: 10.1016/j.biosystemseng.2018.06.017.

[18] M. Dyrmann, H. Karstoft, and H. S. Midtiby, "Plant species classification using deep convolutional neural network," *Biosystems Engineering*, vol. 151, pp. 72–80, Nov. 2016, doi: 10.1016/j.biosystemseng.2016.08.024.

[19] A. A. Mosslah and A. Case, "Telemedicine medical image compression based on ROI (a case study of spine medical images)," *Journal of Global Pharma Technology*, vol. 10, no. 3, pp. 184–190, 2018.

[20] A. A. Mosslah and R. H. Mahdi, "3DMM fitting for 3D face reconstruction," *Journal of Engineering and Applied Sciences*, vol. 13, no. 24, pp. 10482–10489, 2018, doi: 10.3923/jeasci.2018.10482.10489.

[21] T. M. Giselsson, R. N. Jørgensen, P. K. Jensen, M. Dyrmann, and H. S. Midtiby, "A public image database for benchmark of plant seedling classification algorithms," *arXiv Prepr. arXiv1711.05458*, Nov. 2017.

[22] K. A. Fitzgerald, J. A. Fitzgerald, and A. J. Bytheway, "Diabetes information retrieval research," *Journal of Health Informatics in Africa*, vol. 4, no. 2, 2017.

[23] W. Li, X. Ye, Y. Huang, and S. Mahmoodi, "Adaptive dimensional learning with a tolerance framework for the differential evolution algorithm," *Complex System Modeling and Simulation*, vol. 2, no. 1, pp. 59–77, Mar. 2022, doi: 10.23919/CSMS.2022.0001.

[24] A. Subeesh *et al*., "Deep convolutional neural network models for weed detection in polyhouse grown bell peppers," *Artificial Intelligence in Agriculture*, vol. 6, pp. 47–54, 2022, doi: 10.1016/j.aiia.2022.01.002.

BIOGRAPHIES OF AUTHORS

Abd Abrahim Mosslah who was born in the Alaesawi village of Fallujah, obtained his M.Sc. in Computer Science from the College of the University of Mustanseriah, Baghdad, Iraq. He is currently an instructor at the Islamic University of Anbar, Iraq. His research interests are deep learning, machine learning, artificial neural networks, computer networks, image processing, and genetic algorithms. He can be contacted at email: aisl.abide@uoanbar.edu.iq.

Reyadh Hazim Mahdi is currently instructor of College of Science University of Mustanseriah Iraq-Baghdad. His research in computer networks, image processing, and software engineering. He can be contacted at email: reyadhhazim@uomustansiriah.edu.iq.

Hassan Kassim Albahadily obtained Ph.D. from BSUIR in Belarus, computer Science Department University of Mustanseriah IRAQ-BAGHDAD. Area of Research in multimedia compression/encryption, and NLP. He can be contacted at email: hassan@uomustansiriyah.edu.iq.