Knowledge-Based Systems 292 (2024) 111586

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Transfer and zero-shot learning for scalable weed detection and classification in UAV images

Nicolas Belissent [a,*], José M. Peña [b], Gustavo A. Mesías-Ruiz [b,c], John Shawe-Taylor [a], María Pérez-Ortiz [a]

[a] *AI Centre, University College London, London, UK*
[b] *Tec4AgrO group, Institute of Agricultural Sciences, ICA-CSIC, Madrid, Spain*
[c] *Escuela Técnica Superior de Ingeniería Agronómica, Alimentaria y de Biosistemas (ETSIAAB), Universidad Politécnica de Madrid, Madrid, Spain*

ARTICLE INFO

Keywords:
Site-specific weed management
Unmanned aerial vehicles
Precision agriculture
Computer vision
Machine learning
Residual networks

ABSTRACT

In an effort to reduce pesticide use, agronomists and computer scientists have joined forces to develop site-specific weed detection and classification systems. These systems aim to recognize and locate weed species within a crop field, using precision equipment to apply required herbicides timely and only where needed, with the objective of reducing the sprayable surface required to eliminate the given weed and protect the crop, with both economic and environmental benefits. Yet, with climate change on the rise, common weeds are expected to undergo some changes to adapt to their environment, possibly with new or invasive weeds spreading to areas where they did not exist before. These changes (often morphological) as well as new invasions need to be taken into account by future classifiers and detection algorithms to ensure system robustness and adaptation to new habitats/climate dynamics. This paper proposes a set of experiments evaluating the use of transfer learning and zero-shot learning for weed classification using our novel TomatoWeeds dataset. Residual networks of variable depth, pretrained on the Imagenet and/or DeepWeeds datasets were evaluated. A ResNet50 pretrained on both datasets and fine-tuned on the TomatoWeeds dataset performed best, returning a holdout set accuracy of 77.8%, showing the advantageous use of transfer learning in this domain. Zero-shot learning, using both embeddings of images and morphological and habitat text-based descriptions, is implemented to test the ability of machine learning pipelines of recognizing unseen classes at test time (which may arise e.g. due to changing climate dynamics), a learning task in which the field (and our experiments) are still far from satisfactory results. Further research could benefit from larger weed-specific datasets for transfer learning as well as deeper network architectures to improve model performance. The projection-based ZSL could also benefit from larger datasets and new zero-shot learning architectures in hope that unseen classes are accurately projected.

1. Introduction

The COVID-19 pandemic demonstrated the fragility of food supply chains worldwide. In order to be resilient to future crises, the EU has implemented a *Farm 2 Fork strategy* (F2F) where food systems were planned to be redesigned to be more socially, economically, and environmentally sustainable.

A 2019 study estimated that out of the 2 million tonnes of pesticides used worldwide, 47.5% of them were herbicides [1]. According to the European Union statistics office, the consumption of herbicides has not seen a decrease since 2011; analysis suggests that the number has stayed constant to approximately 350 000 tonnes a year [2]. As a result, the EU has implemented a ‘pesticide and herbicide plan’ within the F2F strategy that aims to reduce the use of pesticides and herbicides by 50% by 2030. However, this is no simple task. Depending on their emergence time, density and species, weeds can result in total yield loss if left uncontrolled [3].

In order to reduce the use of pesticides, research has suggested the use of Site-Specific Weed Management (SSWM) where the aim is to reduce the amount of herbicide used by determining the location of the weeds at an early growth stage [4]. SSWM leverages computer vision and deep learning improvements to detect and classify weeds. Ground, drone, and satellite images were used to train image recognition models [5,6]. Once detected, precision agriculture techniques were used to precisely apply herbicide on the weed, reducing drastically the amount of product used. As consequences of climate change, plants were expected to evolve to better suit their environments [7]. Future

[*] Corresponding author.
E-mail address: ucabnb2@ucl.ac.uk (N. Belissent).

https://doi.org/10.1016/j.knosys.2024.111586
Received 24 August 2023; Received in revised form 12 February 2024; Accepted 29 February 2024
Available online 13 March 2024

SSWM methods must be more robust and adaptive in order to detect these changes and maintain accurate classification.

Weed mapping is an essential step to apply site-specific weed control strategies in the context of precision agriculture. Current machine learning models for this purpose utilize various neural network architectures for computer vision applications, trained using specifically tailored weed datasets (with low generalization beyond the specific conditions of the training set), and with high associated labeling and computational costs, all of which precludes broader uptake of this technology and its use for building more sustainable pipelines. To ensure the reproducibility of methods in real cropfield scenarios, it is crucial to build algorithms that can recognize weeds in the most accurate and cost/time-effective manner. This means detecting and classifying weeds seen or unseen at training, at early plant growth stages (to optimize the application of control measurement), but also leveraging multiple low-cost scalable datasets to ensure broad generalization.

This study aims to explore methods that make weed detection models more reproducible, effective and robust. More precisely, transfer learning and zero-shot learning (ZSL) configurations were evaluated using various open-source datasets along with our novel TomatoWeeds dataset. This dataset was generated with a small number of drone-based weed images obtained of a commercial tomato field naturally infested with three different species: *Cyperus rotundus*, *Solanum nigrum*, and *Portulaca oleracea*. We thus propose high performing models trained on an early stage weed dataset, and evaluate how transfer learning may help weed detection task in an effort to reduce computation and labeling cost for future models. Performance of zero-shot learning was also analyzed in order to be further integrated into weed detection algorithms capable of detecting new weeds not previously seen in the target scenario.

2. Materials and methods

2.1. Weed detection algorithms

The global growing awareness towards healthy and more sustainable foods, as well as an increase in cost of labor, have shed light on the benefits of automatic weed control. Automated weed control systems involve detecting the location and species of a weed in a field. For the past decade a variety of weed detection studies have been published. Both machine learning (ML) and deep learning (DL) approaches have been used to solve the detection and classification tasks. Despite good results with classical machine learning techniques (SVM [8,9], LDA [10], K-Means [11]), advancements in Deep Learning methods have helped weed detection achieve higher performing methods and models. These methods have been used to solve different weed detection objectives.

Deep learning has been used for multi-label classification on an image level. The presence is a weed along with its class is given for an input image. Convolutional Neural network (CNN) type architectures are implemented to extract features from these images. Using the DeepWeeds dataset, past studies have used pretrained (Imagenet) Resnet50 architectures [12] as well as a graph-based RNN architecture [13] to perform such classification. The Resnet50 proposed by Olsen et al. [12] achieves considerably high results on the DeepWeeds dataset; reaching a validation accuracy of 95.7%. The graph-based approach achieves an even better accuracy of 98.1%. Peteinatos et al. [14] uses the same Resnet50 pretrained on Imagenet but on a different UAV images of maize, sunflower, and potatoes fields; model performance reaches an accuracy of 97%. Veeranampalayam Sivakumar et al. [15] uses a region based faster CNN and Single-shot Detector (SSD) architectures to perform real-time crop and weed detection on UAV data. Both models were able to predicted accurately (85% and 84% respectively) and promptly. A more recent combines a CNN with learning vector quantization on an UAV dataset where masks have been applied to input images to remove background [16]; model performance reaches 99.44%.

Table 1
Summary of CNN-based weed detection models.

Citation	Model	Accuracy (%)
[12]	Resnet50 (Imagenet)	95.7
[13]	Graph-based RNN	98.1
[14]	Resnet50 (Imagenet)	97
[15]	Faster CNN and SSD	85
[16]	CNN with Learning Vector Quantization	99.44
[17]	YOLO-v3	–
[18]	YOLO (CNN)	0.94 (F_1 Score)
[19]	YOLO-v3, Centernet, Faster R-CNN	0.97 (F_1 Score)
[20]	SegNet	0.8 (F_1 Score)
[21]	UNet	83.23
[22]	Fully Convolutional Network	92.3

To further this objective, some models have the capacity to detect the location of the weed on top of its classification. A rather simplistic YOLO-v3 model for UAV footage was built to detect a single plant class and build a bounding box around the detected plant [17]. Puerto et al. [18] uses a CNN approach with a YOLO (You Only Look Once) architecture for multi-spectral crop row and weed detection. A recently published study compares YOLO-v3, Centernet, and Faster R-CNN for real-time bounding-box weed detection [19]. YOLO-v3 demonstrated the highest accuracy and computational efficiency.

Some models aim to classify weeds and crops on a pixel level; otherwise known as semantic segmentation. Such deep learning models take images as inputs and output an image of labeled pixels. The networks used in such tasks are derived from CNNs as they contains convolutional layers. However, they do not the same architectures given that the output dimensions must match the input dimensions. A SegNet trained on a UAV data classified weeds, crops and background to a high accuracy [20]. Brilhador et al. [21] explores the effects of different data augmentation combinations when performing pixel-wise classification; the designed UNet reports an optimal pixel-accuracy of 83.23% when augmenting images with vertical and horizontal flips. Unlike the latter two models, where the models contain encoder–decoder architectures, Huang et al. [22] uses a pretrained fully convolutional network that is capable of predicting dense class map of UAV rice paddy images; model performance reaches a pixel-accuracy of 92.3% Table 1.

The majority of the research papers suggest models for detecting and/or classifying weeds. These models have typically undergone testing on datasets specifically designed for weeds, comprising images of early growth stage weeds spanning various weed species. This approach aligns with the methodology employed in this paper.

2.1.1. Limitations of weed detection

The existing literature on weed detection has presented effective models, but there is still room for improvement, especially in reducing computation time constraints for real-time applications to maximize crop yield and minimize costs. Weed datasets vary significantly in species, atmospheric conditions, growth stages, resolution, scalability, and acquisition methods, potentially leading high-performing models to be overfitted to specific datasets. To address this, future research should focus on creating a comprehensive dataset covering diverse species, geographies, and atmospheric conditions. The cost of labeling images, whether at the image or pixel level, is a significant challenge, and efforts should be directed towards developing cost-effective labeling methods, such as weakly-supervised, semi-supervised, and unsupervised approaches. Moreover, as climate change affects weed dynamics, life cycles, and geographic ranges, algorithms need to adapt to these changes. Zero-shot learning, a machine learning paradigm, can be a valuable tool in addressing these challenges by enabling models to recognize new weeds with limited or no data.

2.2. Zero-shot learning paradigm

The majority of machine learning models classify data whose classes were seen at training. To cope with ever changing weed species, models must be able to detect new different weeds, that were not present at training. The zero-shot learning paradigm is designed to distinguish seen classes from potential unseen classes.

Zero-shot learning methods can be broken down into two section: classifier-based methods and instance-based methods [23]. Classifier-based methods focus on directly learning a classifier that can detect unseen classes. Instance-based methods, aim to determine a way to generate labels for an unseen testing instances and then use them in a classifier. In this study, the aim is to use an Instance-based method to apply ZSL to a weed detection task. Instance-based methods can be broken down into projection, instance-borrowing and generative methods, as shown in Wang et al. [23], Pourpanah et al. [24].

Generative approaches in zero-shot learning, as explored in recent research, focus on synthesizing visual features for unseen categories. One approach introduces novel fusion techniques at the attribute, feature, and cross-levels, achieving state-of-the-art results on three zero-shot image classification benchmarks, along with successful generalization to zero-shot detection on the MS COCO dataset [25]. Another study proposes a dual generation network framework, leveraging discriminative information from visual features, resulting in superior performance on six benchmark datasets [26]. Both approaches contribute to narrowing the gap between seen and unseen classes in zero-shot learning scenarios, showcasing the potential of generative models in addressing ZSL. Given the complexity of generative approaches, this research opted for the more interpretable projection-based methods.

Projection methods project feature space instances (seen classes) and the unseen prototypes into a common projection space. This will allow to obtain labels instances of unseen classes and build a classifier for them. The advantage of these methods is that the choice of projection function is flexible. It can be chosen to suit the task and dataset at hand. However, given that each unseen class has one labeled prototype, suitable classification algorithms can be limited. Xie et al. [27] proposed a simple ZSL architecture that uses class attributes to build its semantic space. Before word embedding models gained in popularity, some research proposed projection-based architectures that use text-keyword semantic spaces alongside with TF-IDF histograms [28]. Wang et al. [29] proposed a projection-based ZSL implementation that uses CNN extracted instance features, Word2Vec extracted label features and a linear mapping function. Xian et al. [30] was then introduced to incorporate nonlinearity to the model, by incorporating latent variables for every image-class pair. Morgado and Vasconcelos [31] uses a label-embedding semantic space and CNN feature extraction to create a ZSL method based on the complementarity found between class and semantic supervision.

Zero-shot learning methods have never been used in the context of weed detection; highlighting the novelty of our research.

2.3. TomatoWeeds dataset

The novel dataset used and presented in this study consists of two mosaic images ($10521 \cdot 10521$ pixel resolution) of tomato fields in Badajoz, Spain taken at midday on a clear day. This was done to minimize cloud and crop-induced shade. The mosaic images were constructed using a commercial softwwere (Photoscan agisoft) that generates the orthomosaic in a semi-automatic way following a chain of phases. Both images were manually labeled by two experts in weed identification, according to their species using GIS coordinates. The images were taken when the weeds were in early growth stages, which is the right time to apply a control treatment to prevent competition with the crop or further spread across the field. The identified species were labeled as: *C. rotundus*, *S. nigrum* and *P. oleracea*. Fig. 1.

Table 2
Class occurrences in the TomatoWeeds dataset.

Development set			Holdout set
Class	Training set	Validation set	
C. rotundus	1062	361	302
Negative	1434	254	425
P. oleracea	52	71	85
S. nigrum	320	36	38

The data is acquired using a Microdrones md4-1000 UAV mounted with a Sony alpha 6300 point-and-shoot camera. The camera shoots in a visible-range camera and acquires 24.2-megapixel images in the visible RGB spectrum. The camera was equipped with a 19 mm focal length lens. The images were acquired at an altitude of 10 m.

2.3.1. Dataset preparation

As mentioned above, the weeds were labeled by class using GPS coordinates. In order to use these labels in a deep learning setting they must be projected onto the image. To do so, the label coordinates were projected onto the image reference frame and transformed to represent pixel locations. This was done using the QGIS software along with both *shp* and *rasterio* python packages (see Fig. 2).

Once the labels were mapped onto the mosaic image, the mosaic image was split in 2 different parts where 70% of the image would be used for training (development set) and the remaining 30% would be used for testing (holdout set). Performing this split earlier ensures that the there is no overlap between both development and holdout dataset.

The next step was to split the mosaic images into smaller input images for a deep learning classification model. To do so, a sliding square window of dimension $n \cdot n$ is applied to the mosaic. The window slides across the images with an overlap, checking if a label lies in its inner circle (see Fig). The inner circle is defined by a circle of radius R, where $R = n - \gamma$ and γ is a user inputted margin value. For the case of this study, the margin is set to $\gamma = \frac{1/5 \cdot n}{2}$, leaving a fifth of the image out of the circle. This ensures that the labeled weeds remain somewhat central in the image. The images were therefore annotated on a image-level. For simplicity, windows that contain 2 or more weed species were removed; hence multi-label classification is not considered (see Fig. 3).

Given the small size and sparsity of weeds, the number of background images is considerably higher that the labeled images. Negative sampling is used to level out this imbalance; the number of background and labeled images were made equal for both training and holdout sets. However, there is still class imbalance given that there were non uniform amounts of weed species in the field. Some species were more common than others.

The choice of n can be crucial in terms of dataset size and model performance. After discussing with the data provider, it was agreed that the algorithm should take as input an image that would represent a surface smaller than 1 m^2. 64-pixel and 128-pixel sizes both represent a surface werea of 32 and 64 cm^2. These dimensions will be tested in further experiments.

A summary of class occurrences across training, validation and holdout sets is given in Table 2.

2.4. Weed detection and classification methods

A ResNet architecture was used to detect and classify weeds. This architecture leverages skip connections to avoid the vanishing gradient problem. They have shown very good performance in image classification tasks, especially in the field of weed classification [12,15].

This study evaluates the performance of different ResNet architectures and transfer learning on a weed detection and classification task for the TomatoWeeds dataset. For transfer learning, the whole model is finetuned.

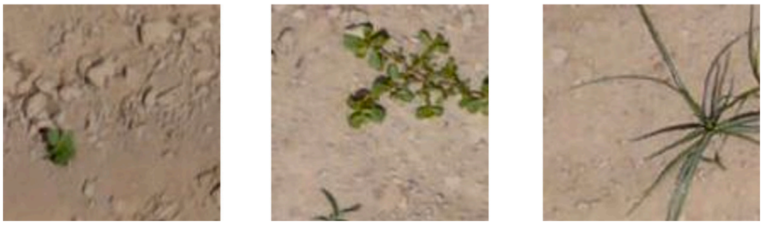

Fig. 1. *S. nigrum*, *P. oleracea* and *C. rotundus* weeds (from left to right) at early stages.

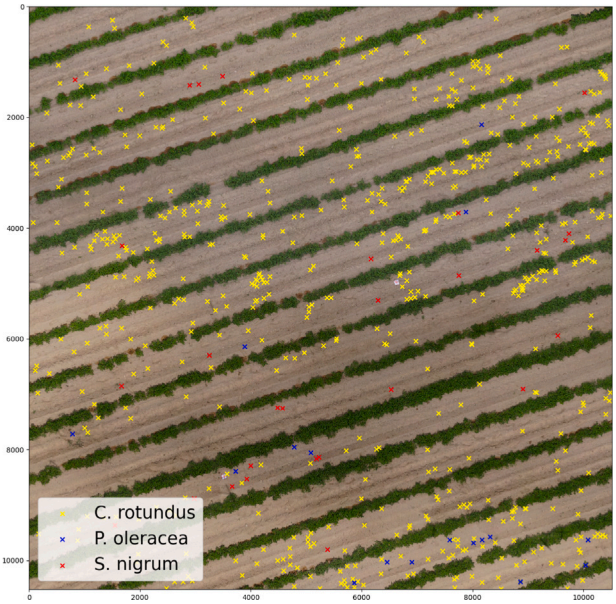

Fig. 2. Entire mosaic image of the TomatoWeeds dataset.

2.4.1. Learning setting and performance metrics

A development containing 70% of the dataset was broken into a training and validation splits, following a 80/20 split. The left 30% of the dataset was used as a holdout set, for testing purposes. Models were all trained on the training set for 100 epochs using the cross-entropy loss function [32] and Adam optimizer [33].

In order to optimize learning rate and subsequent model performance, a step-wise learning rate scheduler was used. Such schedulers, help the model approach the optimum more efficiently. In practice, they apply different types of decay functions to the learning rate; meaning that as the model approaches an optimum, the learning rate will decrease and allow for more precise updates.

All models were evaluated using the accuracy metric. This metric corresponds to ratio of all correctly classified images against the total images. Despite often providing with an accurate representation of model performance, accuracy is often not a suitable metric for imbalanced classification problems. Thus, recall and precision metrics were also used to report performance across all classes.

Recall is measure of how many correct class-specific predictions over all the occurrences of that class. Precision is a measure of many correct class-specific predictions were over all the prediction made for that class. Described often as the harmonic mean of both metrics, f1 score is used to encapsulate both recall and precision metrics, in a balanced way. As shown below, in order to obtain high f1 score, recall and precision values must be high.

$$\text{F1 Score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$

Table 3
Class occurrences in the original DeepWeeds dataset.

Class	Entire dataset	ZSL Training set	ZSL Test set
Chinee apple	1125	900	225
Lantana	1064	852	212
Parkinsonia	1031	825	206
Parthenium	1022	818	204
Siam weed	1074	860	214
Snake weed	1016	813	203
Prickly acacia	1062	–	250
Rubber vine	1009	–	250
Negative	9106	–	–
Total	17 509	5068	1764

2.5. Zero-shot learning for unseen weed classification

Zero-shot learning has been applied to various different image-based tasks in literature. However, its application to weed classification is novel. Suggested in the 2019 Weed detection survey [34], ZSL has since not been mentioned in any SSWM research. Hence, this section introduces a novel approach to weed detection and classification. The second part of our experimental design aims to test ZSL in the domain of weed classification.

The real-world application of ZSL in this domain aims to flag to agronomists (or any other end user) when a weed that is not contained in the training set has appeared in the field. The proposed system could provide a classification to an unknown class. The agronomist can then verify the validity of this prediction and add the new plant to the training set. This, which can commonly be done through uncertainty quantification techniques [35]. ZSL aims to include the information into the classification system in such a way that when the same weed emerges it can be associated to this same category seen before, without requiring explicit training.

2.5.1. DeepWeeds dataset for zero-shot learning

The previously described TomatoWeeds dataset contains 4232 images across four imbalanced classes. Hence, to ensure the best results possible for this novel approach, we focus our ZSL experiments on the DeepWeeds dataset [12]. It has a large number of balanced classes with many images per class. The negative background class was not considered in these experiments as the aim was to detect and classify new weeds.

The dataset was manipulated to fit the ZSL task; we drop two classes from the training set, so that unseen classes are only present in the test set and we can monitor the performance on those classes. In order to maintain the test set class-wise balance, 250 random samples of both unseen classes were retained. Class-wise balance was required to ensure that the unseen classes were fairly represented when projecting them into feature spaces, later used for classification. Table 3 summarize these changes.

2.5.2. Semantic spaces used for ZSL

Many ZSL approaches rely on projection spaces and notions of distances to classify unseen classes [23]. Such projection methods aim to project both seen and unseen instances into a common projection

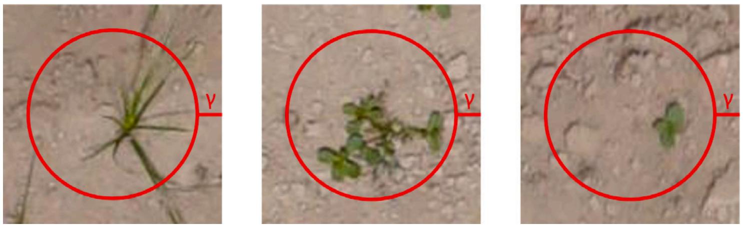

Fig. 3. Visualizing the *inner circle* splitting protocol for *C. rotundus*, *P. oleracea* and *S. nigrum* (from left to right).

Table 4
Examples of morphological and habitat descriptions [36].

Weed type	Morphological description	Habitat description
Chinee Apple	Small, hairless trees up to 10 m tall. Branches were slender, zigzag shaped, with sharp spines. Leaves have a short, spine-tipped stalk. Leaf branches were 20–40 cm long. Flowers were yellow, fragrant, 5-petaled, each on a long, slender, drooping stalk. Seed pods were pencil-like, 5–10 cm long, constricted between seeds. Seeds were oval, about 15 mm long, with thick, extremely hard coats.	Occurs most abundantly on flood plains but adaptable to a wide range of soil types. Found along watercourses in sub-humid and semi-arid wereas of Queensland.

or semantic space, which is built to maximize class discrimination and used to detect occurrences of new unseen classes. In this work, we experiment with both image and text semantic spaces, as we observe in our initial experiments that image spaces are not enough to do ZSL of weed classification.

Image-representation space. Image-representation semantic spaces were built by projecting images into a vector space. This can be done using the output layer of a neural network. The backbone of ResNet50 was used to build image-representation embeddings. Given that the model produced some of the most accurate results, the output layer would discriminate the most between classes; therefore aiding further analysis. The output layer of the ResNet50 model is a $1 \cdot 2048$ vector, meaning the projection space would be 2048-dimensional. This high-dimensional space may be reduced to maximize class variability using dimension reduction techniques.

Text-embedding space. Apart from the images available, we extracted morphological and habitat descriptions of seen classes using web scraping techniques on a Queensland Government website [36]. An example of description for a class can be seen in Table 4.

Text-embedding spaces were built using seen and unseen class descriptions and word embedding models; terms in the descriptions were converted to vectors that provide a latent representation of the seen classes. Word embedding models leverage natural language processing techniques to project words into a vector space or semantic space.

Both morphological and habitat descriptions were separately preprocessed before being projected into latent space and concatenated. This preprocessing involved removing stop words and performing stemming and lemmatization. The remaining words, $\mathcal{W}$, were then individually converted to vectors, $\vec{v}_t$, using the *glove-25-twitter* word embedding model [37]. The average term embedding, $\vec{v}_{avg}$, was computed as follows,

$$\vec{v}_{avg} = \frac{1}{|\mathcal{W}|} \sum_{t_{\mathcal{W}} \in \mathcal{W}} \frac{\vec{v}_{t_{\mathcal{W}}}}{\left\| \vec{v}_{t_{\mathcal{W}}} \right\|}.$$

Both morphological and habitat descriptions were used to create two average term embeddings, using the equation above. These embeddings were then concatenated to create a $1 \cdot 50$ vector; therefore creating a 50-dimensional latent semantic space. This concatenation is illustrated in Fig. 4. The dimensionality of this embedding may be reduced to ease further analysis.

We hypothesize (and visually observe in our embeddings) that the benefit of text-embedding semantic spaces is that text descriptions are more nuanced and often better at discriminating between classes when images show little inter-class variability.

2.6. Image to text projection

As mentioned above, we suspect that embeddings of text descriptions of weeds would represent class-wise difference better than weed images themselves. To explore this in more detail, we use inspiration from the literature [38,39], where it is shown that we can project from images to text, and use this new representation, combined with the scrapped morphological and habitats descriptions, as a suitable semantic space for class discrimination and ultimately ZSL.

Specifically, we use multi-dimensional regressions (MDR) to draw a relationship between two multi-dimensional spaces. In this study, the MDR is built using a neural network with an encoder–decoder architecture. This architecture can be broken down into the three following parts:

- **Encoder:** A module that compresses the input into a much smaller representation using a feed-forward neural network.
- **Decoder:** A module that up samples the smaller representation to match the output dimension. In practice, the decoder projects to a lower dimensional subspace that matches the latent label space.
- **Bottleneck:** A module used to restrict the flow of information between both encoder and decoder modules [40]. This helps form a knowledge-representation of the input, where only vital information about the inputs is shared with the decoder. In practice, this element enables the models to project in a more class-discriminative manner, as it outperforms standard feed-forward architectures.

This study uses this architecture to learn a projection function that takes as input an image embedding and outputs a text embedding; the 2048-dimensional inputs were projected onto a 50-dimensional label space. The network architecture, illustrated in Fig. 5, contains a 7 layer encoder, a 2 layer decoder and 16-dimensional bottleneck.

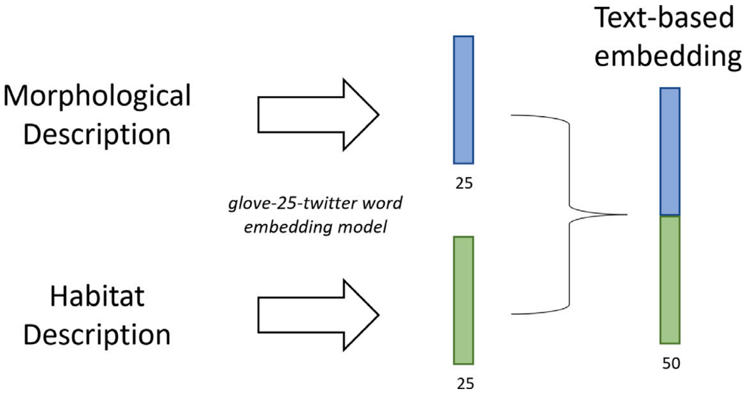
Morphological Description
Habitat Description
glove-25-twitter word embedding model
25
25
Text-based embedding
50

Fig. 4. Creating text embeddings from both morphological and habitat weed descriptions.

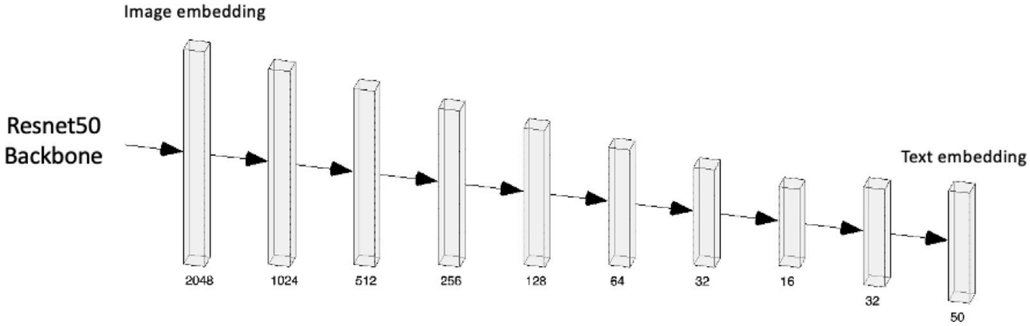

Fig. 5. Architecture of the multi-dimensional regression used to build a projection-based semantic space.

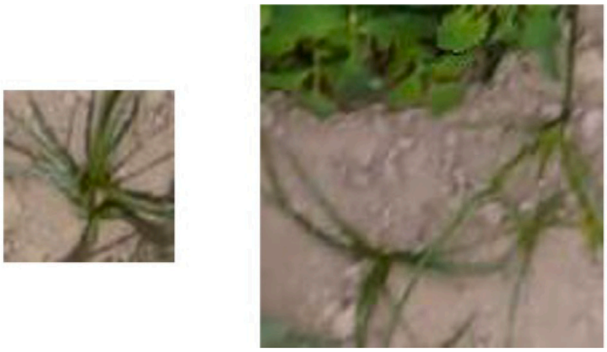

Fig. 6. 64-pixel (left) and 128-pixel (right) input images for the *C. rotundus* weed species.

This architecture is trained using a mean-squared error loss function (MSE). MSE loss aims to reduce the distance between predicted embeddings, p_i, and target embeddings, t_i, by applying the following function, $\mathcal{L}(p_i, t_i) = \sum_{i=1}^{D} (p_i - t_i)^2$.

3. Empirical results

3.1. Weed classification and transfer learning experiments

First, we aim to test how the size of images impact our weed detection and classification pipelines. Two datasets created with 64-pixel and 128-pixel input images were tested across ResNet models of variable depths. Open-source ResNet models pretrained on Imagenet [41] were used to perform this experiment. Fig. 6 depicts the different image sizes.

In parallel with the split size evaluation, the effect of a varying model depth was assessed. ResNet18, ResNet50, and ResNet152 models were applied to both datasets. These models have respectively 18, 50, and 152 layers. The same open-source ResNet models were used to perform this experiment. The results of these two tests can be seen in Tables 5 and 6.

Table 5
ResNet performance both 64-pixel and 128-pixel image splits.

Model	128-pixel split	64-pixel split
ResNet18	0.327	0.697
ResNet50	0.377	0.776
ResNet152	0.340	0.779

Finally, the effectiveness of transfer learning is evaluated by testing model performance across various pretraining configurations. Open-source ResNet models pretrained on Imagenet and ResNets manually pretrained on the DeepWeeds dataset [12] were tested across different configurations; seen in Table 6.

When using the trained ResNet50 model and the test set specified in the DeepWeeds study [12], the model yielded a 95.7% weighted accuracy; perfectly matching the results obtained in the study. The baseline was deemed replicated.

All metrics shown below were issued from the holdout set, a dataset completely independent of both training and validation sets. Hence, the results give a good idea of model generalization beyond the training set.

The tables below demonstrate clearly how a 64-pixel split is better suited to this classification task. The larger ResNet152 shows poor performance for a dataset with a 128-pixel split size; yielding an accuracy of 0.34. The same model performs considerably better on the dataset with a 64-pixel split size; yielding an accuracy of 0.78.

In addition, these tables demonstrate that deeper networks perform better for this weed classification task; ResNet50 and ResNet152 model performs well for both datasets.

Table 5 shows how model accuracy is affected by different pretraining configurations. The first configuration evaluates model performance

Table 6
Model performance (holdout set accuracy) depending on pretraining configuration; PT for pretrained, FT for finetuned.

Configuration	ResNet50	ResNet152
No PT /trained from scratch on TomatoWeeds	0.737	0.497
PT on Imagenet + finetuned on TomatoWeeds	0.776	0.781
PT on Imagenet + DeepWeeds + FT on TomatoWeeds	0.778	–
PT on Imagenet + DeepWeeds + no FT	0.486	–

Table 7
Class-wise precision, recall and F1 for both best performing models.

Model	ResNet50			ResNet152		
	Precision	Recall	F1	Precision	Recall	F1
C. rotundus	0.75	0.83	0.79	0.73	0.81	0.77
P. oleracea	0.69	0.32	0.44	0.67	0.49	0.57
S. nigrum	0.29	0.45	0.35	0.37	0.37	0.37
Negative	0.88	0.87	0.87	0.87	0.85	0.86

Table 8
Evaluation of the two embeddings (image and text) for within cluster and between cluster metrics.

Projection evaluation metric	Image-based projection	Text-based projection
WCSS	5284.9	1033.8
BCSS	5.73	11.47

when no pretraining was applied. This entailed training both ResNets from scratch on the TomatoWeeds dataset. The shallower ResNet50 was able to reach a fair accuracy of 0.73, whereas the deeper ResNet152 failed to converge to high performing state, returning an accuracy of 0.49.

Then, a simple fine-tuning experiment was led, where the models were both initially pretrained in the Imagenet dataset. They were then finetuned on the TomatoWeeds dataset. Both models produced accurate results; 0.77 and 0.78 for both ResNet50 and ResNet152 models. These results demonstrate the benefits of finetuning on larger datasets.

The final two experiments involved fine-tuning on the larger DeepWeeds dataset [12]. Given the size of this dataset and limited computational resources, the experiments were only able to be tested on the smaller ResNet50 model.

As expected, pretraining on Imagenet and on the larger weed-specific DeepWeeds dataset before finetuning on TomatoWeeds aided performance on the TomatoWeeds holdout set. However, this improvement is marginal. This could potentially be because DeepWeeds images were taken on ground for different crop and weed species, rather than with unmanned aerial vehicles.

When finetuning on TomatoWeeds is not considered and only the pretrained Imagenet and Deepweeds model were used on the TomatoWeeds hold out set, results were poor; yielding a hold out accuracy of 0.49. Despite showing poorer performance, the latter model performed better than a no skill 4-class classifier; demonstrating potential weed feature knowledge transfer between both Deepweeds and TomatoWeeds datasets.

Throughout this series of pretraining and fine-tuning experiments, we could conclude that pretraining a model on a larger dataset prior to fine-tuning it on a smaller more specific dataset, will aid holdout set performance on this smaller dataset. This could significantly improve weed mapping pipelines, where the bottleneck is the collection of the field images, meaning potentially we can achieve better detection in a more cost-efficient manner by reusing previous weed mapping datasets (even if they contain different species), and even when other large-scale object detection datasets like Imagenet are used (which may help to train features that perform well across different computer vision tasks, e.g. edge detectors).

As demonstrated in the experiments above, the best performing models were the ResNet50 pretrained on both datasets and the ResNet 152 only finetuned on Imagenet. Both models perform similarly across all classes. Fig. 7 illustrates model convergence with a step-wise learning rate scheduler.

C. rotundus and background were considered as majority classes as they have more instances within the entire dataset. On the other hand, *P. oleracea* and *S. nigrum* appear considerably less. This class imbalance is reflected in the class-wise performance of both models. The majority classes produce good recall and precision metrics, reflected in a good f1 score; returning an average score of 0.78 and 0.87 for *C. rotundus* and background classes across both models. The minority classes produce considerably lower F1 scores. Both models return the same score of 0.37 for the *S. nigrum* class. However, for the *P. oleracea* class, the deeper ResNet152 model performs slightly better, returning a score of 0.57 Table 7.

3.2. Zero-shot learning experiments

As mentioned, both image-based and text-based semantic spaces were used to classify seen and unseen classes, as only using image-based embeddings was not shown satisfactory. Our aim was to project images into a space where potentially inter-class variance was higher; ideally improving chances of higher performance. Fig. 8 illustrates the differences between image-based and text-based projection methods.

In order to further the comparison between both text-based and image-based projection spaces, we analyzed the cluster characteristics for the two dimensional projection shown above. We evaluated the following below.

1. **Within-Cluster Sum of Squares (WCSS)** represents the sum of squared distances between each data point and the centroid of its assigned class.
2. **Between-Cluster Sum of Squares (BCSS)** represents the sum of squared distances between the centroids of different classes and the centroid of all data points. It measures the dispersion between classes.

To provide a fair comparison between both projection methods, the embeddings were both normalized using z-score normalization. Given their high dimensions and exponentially growing distances, the embeddings were reduced to 2D using PCA to get a more interpretable evaluation metric.

As shown in Table 8, the text-based projection demonstrates superior discrimination of classes compared to the image-based projection. This is evident from the significantly lower WCSS value of the text-based projection, indicating that the classes are more tightly packed and distinct. In addition, the image-based projection exhibits a lower BCSS value, suggesting smaller separation between classes. This, once again favors the text-based projection, despite being a relatively small difference. Therefore, we have concluded that the text-based projection is more suitable for zero-shot learning (ZSL) purposes.

The DeepWeeds images were converted to image-based embeddings using the CNN backbone of the best-performing ResNet50 model described in Section 3.1; these embeddings can be seen as an image-based space. As mentioned, the text-based space was then constructed using text descriptions of both the morphological aspects and habitats of both seen and unseen weeds. As described in Section 2.5.2, these descriptions were converted into embeddings by leveraging natural language processing techniques. A multi-dimensional regression (MDR) was then used to project image-based into a text-based semantic space, which was later concatenated to the original text space. The projected label embeddings were then classified in a semi-supervised fashion where the ground-truth label embeddings were used as known embedding *centroids*. At training, know text embeddings of both seen and unseen

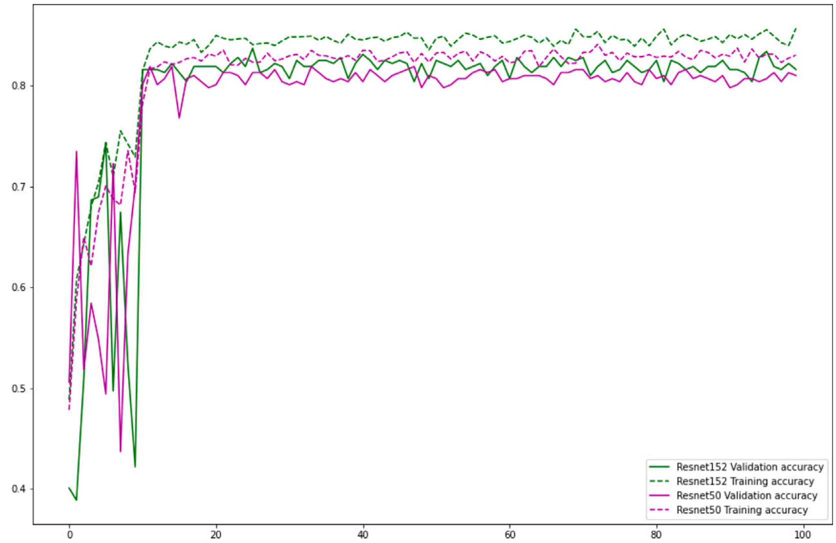

Fig. 7. DeepWeeds pretrained ResNet50 and ResNet152 model convergence; with a step-wise learning rate scheduler.

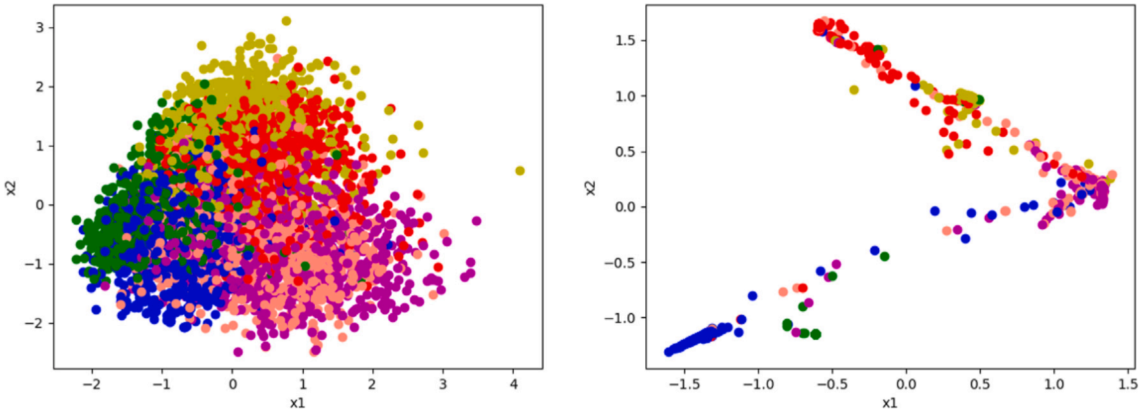

Fig. 8. Image-based (left) and text-based (right) projection reduced to two dimensions using PCA and normalized using z-score normalization.

classes are used to train this MDR. At testing, unseen and seen classes are projected into text-based space where reside known unseen and seen class centroids. The projections are then classified corresponding to their nearest known centroid.

Fig. 9, illustrates this system at testing, where some classes have not been seen before. Given that ground-truth known labels were used to aid classification of unseen class, this experiment can be considered as *Class-Transductive Instance-Inductive* learning. Two zero-shot experiments were derived from this setting.

Testing seen class classifications. Prior to testing this projection-based ZSL architecture on unseen classes, it was crucial to test performance on seen classes exclusively. This experiment is seen as an initial baseline to test this classification architecture.

Classifying the entire dataset. Then, an experiment involved classifying both seen and unseen classes at testing. All instances were projected into the semantic space and both Nearest-centroid and Label propagation algorithms were used to classify these projections. A PCA dimension reduction was used to determine the optimal number of dimensions to retain in order to maximize performance.

Evaluating unseen class projections. To better understand performance on unseen classes a second experiment was led solely on unseen class projections. All seen classes instances were removed from the hold out set. This experiment involved computing the nearest unseen centroid from all unseen projections, within the semantic space. The idea of this experiment is to evaluate how accurate the MDR architecture is able to project unseen classes. Good performance would translate to better than random prediction, where unseen classes were projected nearer to their respective unseen centroids.

3.2.1. *Zero-shot learning results*

Prior to testing this projection-based ZSL architecture on unseen classes, it was crucial to test performance on seen classes exclusively. Both label propagation and nearest centroid performed similarly. When retaining 5 dimensions, they returned overall accuracy, precision and recall scores of 0.77. A class-wise breakdown of the f1-scores can be seen in Table 9. These scores do not match the performance of the ResNet50 model proposed in the original DeepWeeds paper [12]. Nonetheless, they were considered valid for further ZSL testing.

When classifying projections of the entire hold out set, seen and unseen weeds, both nearest centroid and label propagation algorithms perform identically. The classifiers returned their best performance when the embeddings were projected into a 4-dimensional subspace; yielding a total accuracy of 0.55. Both models perform similarly on unseen classes; returning average accuracies of 0.005 for *Prickly acacia* and approximately 0.34 for *Rubber vine*.

Table 9, illustrates the highest class-wise f1 scores and overall accuracies across both algorithms; optimal performance is similar.

Table 9 illustrates mixed results. On one hand, the unseen Rubber Vine class performs to the standard of a seen class, returning an f1 score of 0.37. On the other, the instances of Prickly Acacia were completely

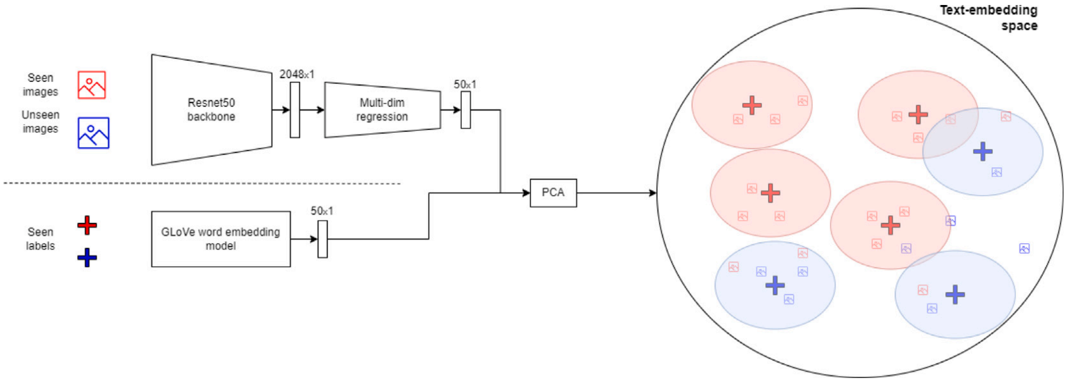

Fig. 9. Representation of our ZSL architecture at testing. The ResNet50 backbone and multidimensional regression were trained prior solely on a set of seen images and labels. The models were expected to transfer knowledge from seen to unseen images. Given the nature of a *Class-Transductive Instance-Inductive* setting, known labels of seen and unseen classes were used to create a text-embedding space. At testing, seen and unseen images were projected onto this space. Images were then classified using label propagation and nearest centroid algorithms.

Table 9
Class-wise F1 Score and average accuracy for ZSL classification methods under the CTII learning setting. Unseen classes were shown in italic.

Class	Baseline	Nearest Centroid	Label propagation
Chinee apple	0.57	0.28	0.14
Lantana	0.84	0.54	0.55
Parkinsonia	0.95	0.52	0.52
Parthenium	0.88	0.32	0.32
Siam weed	0.90	0.60	0.54
Snake weed	0.49	0.14	0.10
Prickly acacia	–	0.00	0.01
Rubber Vine	–	0.30	0.37
Total Accuracy	0.77	0.326	0.326

Table 10
Class-wise distances between respective projected and ground-truth locations within the text-based semantic space; unseen classes shown in *italics*.

Class	Average l2-distance to centroid
Chinee Apple	0.14
Lantana	0.06
Parkinsonia	0.02
Parthenium	0.06
Siam Weed	0.02
Snake Weed	0.17
Prickly Acacia	0.41
Rubber Vine	0.39

misclassified; returning an f1 score of 0.0. An analysis of the latent label space was led in order to develop an understanding for the f1 scores shown in table.

When computing the average class-wise l2-distances from the projected point to the centroids, the difference between seen and unseen projected points is drastic. On average, an unseen class projection is nearly 5 times further from its respective centroid than for a seen class projection; average l2-distances of 0.08 and 0.38 for seen and unseen classes respectively. Table 10 summarizes class-wise l2-distances. Poor projections for unseen classes will inevitably reflect on classification performance.

Finally, when removing all seen classes from the hold out set to focus solely on unseen class projections, the nearest centroid algorithm performs fairly returning an optimal accuracy of 0.648 when 9 dimensions were retained for clustering. In other terms, unseen classes are projected closer to their respective centroids more often than not. This result suggests that the MDR is learning at a relationship between weed images and textual descriptions as it is able to classify unseen weeds at a higher than random rate, which could be used to build a threshold-based distance anomaly detector that could similarly detect unseen classes.

4. Discussion and conclusions

This study explores different methods that allow weed detection models to be more accurate, cost-effective and potentially reproducible in the context of climate change. We first show the potential of transfer learning, even when the pretraining is done with object detection datasets, instead of weed mapping ones. We then conducted ZSL experiments to evaluate whether we can build more robust weed detection models that can detect weed species unseen at training.

Transfer learning for weed detection. A diverse set of ResNet models, coupled with various transfer learning configurations, were tested using the novel TomatoWeeds UAV dataset. Among the models evaluated, the most successful included a ResNet50 pretrained on both Imagenet and DeepWeeds datasets and a ResNet152 fine-tuned solely on Imagenet. The application of transfer learning significantly enhanced the weed classification task's performance. However, despite achieving commendable hold-out set accuracies of 0.77, these models struggled to consistently excel across all classes, particularly with less common weeds. Agronomists on our team deemed these results acceptable, asserting their alignment with a valid performance range, especially in the context of early growth weed detection. The study advocates prioritizing low errors of omission over higher errors of commission, emphasizing the importance of over-detection to prevent leaving weeds undetected.

Many aspects of this study can be discussed to develop a better understanding of the assumptions made and how they have impacted results. This starts at the conception of the dataset. The images, captured at noon under clear atmospheric conditions with minimal shade and noise, introduce a bias in the model due to an almost ideal data acquisition setting. Future research could investigate the effect of noise on UAV data and potentially construct a noise model, enabling data acquisition with fewer constraints. Additionally, the study overlooks the multi-label scenario when processing mosaic images, which could significantly affect scalability depending on weed density in inter-crop rows.

The study also delves into the model itself, noting the performance boost observed in the ResNet50 model when fine-tuned on the DeepWeeds dataset. Limited computational resources prevented testing this approach on the larger ResNet152. Exploring deeper networks might uncover better-discriminating weed features, potentially leading to improved classification performance.

The distribution of classes within the dataset is another crucial aspect. While the overall model performance on the TomatoWeeds dataset met the approval of subject matter experts, the class imbalance within the dataset hinders a comprehensive evaluation, with both classes returning F1 scores of at most 0.57 and 0.37. Strategies for addressing imbalanced classification could enhance the robustness of the model.

Lastly, the study calls for future research to focus on aggregating more weed-specific data across diverse weed species, geographies, and climate conditions to ensure the reproducibility of weed classification models. With larger and more varied weed datasets, subsequent transfer learning experiments can be conducted to push the boundaries of model performance.

Zero-shot learning for weed detection. The experiments conducted in the zero-shot learning (ZSL) setting aimed at enhancing the climate-resilience of weed detection models yielded mixed results. Despite testing various projection methods, dimensions, and semantic spaces, the ZSL approach exhibited limited success, particularly in correctly classifying unseen classes during testing. The insufficiency of data, both in terms of classes and instances, emerged as a significant challenge, suggesting that future endeavors in this direction demand a substantial increase in data availability to ensure more reliable outcomes.

The fusion of weed detection and ZSL, a novel exploration in this study, necessitated multiple assumptions in constructing semantic spaces, projection functions, and embeddings. The use of the relatively compact *glove-twitter-25* word embedding model for constructing the text-based semantic space, while computationally efficient, may have limited the efficacy of the projection function. A potential avenue for improvement could involve exploring larger text-based semantic spaces, although this would require greater computational capacity.

The study's choice of only eight classes for ZSL implementation, with a significant proportion being unseen, deviated from the higher class counts typically employed in ZSL research. Based on existing literature, it is evident that a larger variety of weed species is necessary for ZSL to excel in weed classification tasks. Furthermore, this study employed ZSL methods with limited computational demands, overlooking more recent, high-performing models that utilize generative approaches, such as generative adversarial networks [42,43]. Future research should strive to incorporate these advanced models to potentially achieve superior performance.

In the context of zero-shot learning, where minimal or no information about unseen classes is available during training, a more flexible approach could be explored. Integrating a selection of unseen class instances and labels with the training set, similar to one-shot or few-shot learning settings, may enhance the model's ability to perform well on unseen classes.

Finally, the study raised critical questions regarding the applicability of ZSL in the face of climate change, which is anticipated to alter plant species characteristics. The assumption that changed plants would resemble entirely new classes warrants scrutiny. Plants may change only slightly, not deviating from there original morphology as much as what is assumed. To account for this, the study suggests that models may benefit from online and lifelong learning methods to adapt to evolving plant characteristics over time. A deeper understanding of how weed characteristics evolve is deemed essential for advancing the climate resilience of future weed detection models.

CRediT authorship contribution statement

Nicolas Belissent: Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Conceptualization. **José M. Peña:** Writing – review & editing, Data curation. **Gustavo A. Mesías-Ruiz:** Writing – review & editing, Data curation. **John Shawe-Taylor:** Supervision, Conceptualization. **María Pérez-Ortiz:** Writing – review & editing, Validation, Supervision, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

This work was partially supported by the Spanish Research State Agency (AEI/10.13039/501100011033) and the NextGenerationEU/PRTR through the Projects PDC2021-121537-C22 and PID2020-11322 9RB-C41. The author GM-R has been a beneficiary of a FPI fellowship by the Spanish Ministry of Education and Professional Training (PRE2018-083227).

References

[1] A. Sharma, V. Kumar, B. Shahzad, M. Tanveer, G.P.S. Sidhu, N. Handa, S.K. Kohli, P. Yadav, A.S. Bali, R.D. Parihar, et al., Worldwide pesticide usage and its impacts on ecosystem, SN Appl. Sci. 1 (11) (2019) 1–16.

[2] Eurostat, Pesticide use in europe, 2019, URL https://ec.europa.eu/eurostat/databrowser/view/aei_fm_salpest09/default/map?lang=en.

[3] B.S. Chauhan, Grand challenges in weed management, Front. Agronomy 1 (2020) http://dx.doi.org/10.3389/fagro.2019.00003, URL https://www.frontiersin.org/article/10.3389/fagro.2019.00003.

[4] S. Christensen, H. Søgaard, P. Kudsk, M. Nørremark, I. Lund, E. Nadimi, R. Jørgensen, Site-specific weed control technologies, Weed Res. 49 (2009) 233–241, http://dx.doi.org/10.1111/j.1365-3180.2009.00696.x.

[5] M. Pérez-Ortiz, J. Peña, P. Gutiérrez, J. Torres-Sánchez, C. Hervás-Martínez, F. López-Granados, A semi-supervised system for weed mapping in sunflower crops using unmanned aerial vehicles and a crop row detection method, Appl. Soft Comput. 37 (2015) http://dx.doi.org/10.1016/j.asoc.2015.08.027, URL https://www.sciencedirect.com/science/article/pii/S1568494615005281.

[6] C. Fernández-Quintanilla, J.M. Peña, D. Andújar, J. Dorado, A. Ribeiro, F. López-Granados, Is the current state of the art of weed monitoring suitable for site-specific weed management in arable crops? Weed Res. 58 (4) (2018) 259–272, http://dx.doi.org/10.1111/wre.12307, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/wre.12307, URL https://onlinelibrary.wiley.com/doi/abs/10.1111/wre.12307.

[7] C. Parmesan, M.E. Hanley, Plants and climate change: complexities and surprises, Ann. Botany 116 (6) (2015) 849–864, http://dx.doi.org/10.1093/aob/mcv169, arXiv:https://academic.oup.com/aob/article-pdf/116/6/849/17637271/mcv169.pdf.

[8] S. Murawwat, A. Qureshi, S. Ahmad, Y. Shahid, Weed detection using SVMs, Eng., Technol. Appl. Sci. Res. 8 (2018) 2412–2416, http://dx.doi.org/10.48084/etasr.1647.

[9] Z. Kiala, O. Mutanga, J. Odindi, K. Peerbhay, Feature selection on sentinel-2 multispectral imagery for mapping a landscape infested by parthenium weed, Remote Sens. 11 (16) (2019) http://dx.doi.org/10.3390/rs11161892, URL https://www.mdpi.com/2072-4292/11/16/1892.

[10] A. Wendel, J. Underwood, Self-supervised weed detection in vegetable crops using ground based hyperspectral imaging, in: 2016 IEEE International Conference on Robotics and Automation, ICRA, 2016, pp. 5128–5135, http://dx.doi.org/10.1109/ICRA.2016.7487717.

[11] C. Hung, Z. Xu, S. Sukkarieh, Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a UAV, Remote Sens. 6 (12) (2014) 12037–12054, http://dx.doi.org/10.3390/rs61212037, URL https://www.mdpi.com/2072-4292/6/12/12037.

[12] A. Olsen, D.A. Konovalov, B. Philippa, P. Ridd, J.C. Wood, J. Johns, W. Banks, B. Girgenti, O. Kenny, J. Whinney, et al., DeepWeeds: A multiclass weed species image dataset for deep learning, Sci. Rep. 9 (1) (2019) 1–12.

[13] K. Hu, G. Coleman, S. Zeng, Z. Wang, M. Walsh, Graph weeds net: A graph-based deep learning method for weed recognition, Comput. Electron. Agric. 174 (2020) 105520, http://dx.doi.org/10.1016/j.compag.2020.105520, URL https://www.sciencedirect.com/science/article/pii/S0168169920303458.

[14] G.G. Peteinatos, P. Reichel, J. Karouta, D. Andújar, R. Gerhards, Weed identification in maize, sunflower, and potatoes with the aid of convolutional neural networks, Remote Sens. 12 (24) (2020) http://dx.doi.org/10.3390/rs12244185, URL https://www.mdpi.com/2072-4292/12/24/4185.

[15] A.N. Veeranampalayam Sivakumar, J. Li, S. Scott, E. Psota, A. J. Jhala, J.D. Luck, Y. Shi, Comparison of object detection and patch-based classification deep learning models on mid- to late-season weed detection in UAV imagery, Remote Sens. 12 (13) (2020) http://dx.doi.org/10.3390/rs12132136, URL https://www.mdpi.com/2072-4292/12/13/2136.

[16] M.A. Haq, CNN based automated weed detection system using UAV imagery, Comput. Syst. Sci. Eng. 42 (2) (2022) 837–849.

[17] A. Etienne, D. Saraswat, Machine learning approaches to automate weed detection by UAV based sensors, in: J.A. Thomasson, M. McKee, R.J. Moorhead (Eds.), Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping IV, Vol. 11008, SPIE, 2019, pp. 202–215, http://dx.doi.org/10.1117/12.2520536.

[18] A. Puerto, C. Pedraza, D.A. Jamaica-Tenjo, A. Osorio Delgado, A deep learning approach for weed detection in lettuce crops using multispectral images, AgriEngineering 2 (2020) http://dx.doi.org/10.3390/agriengineering2030032.

[19] X. Jin, Y. Sun, J. Che, M. Bagavathiannan, J. Yu, Y. Chen, A novel deep learning-based method for detection of weeds in vegetables, Pest Manage. Sci. 78 (5) (2022) 1861–1869.

[20] I. Sa, Z. Chen, M. Popović, R. Khanna, F. Liebisch, J. Nieto, R. Siegwart, Weednet: Dense semantic weed classification using multispectral images and MAV for smart farming, IEEE Robot. Autom. Lett. 3 (1) (2018) 588–595, http://dx.doi.org/10.1109/LRA.2017.2774979.

[21] A. Brilhador, M. Gutoski, L.T. Hattori, A. de Souza Inácio, A.E. Lazzaretti, H.S. Lopes, Classification of weeds and crops at the pixel-level using convolutional neural networks and data augmentation, in: 2019 IEEE Latin American Conference on Computational Intelligence (la-CCI), 2019, pp. 1–6, http://dx.doi.org/10.1109/LA-CCI47412.2019.9037044.

[22] H. Huang, J. Deng, Y. Lan, A. Yang, X. Deng, L. Zhang, A fully convolutional network for weed mapping of unmanned aerial vehicle (UAV) imagery, PLOS ONE 13 (4) (2018) 1–19, http://dx.doi.org/10.1371/journal.pone.0196302.

[23] W. Wang, V.W. Zheng, H. Yu, C. Miao, A survey of zero-shot learning: Settings, methods, and applications, ACM Trans. Intell. Syst. Technol. 10 (2) (2019) http://dx.doi.org/10.1145/3293318.

[24] F. Pourpanah, M. Abdar, Y. Luo, X. Zhou, R. Wang, C.P. Lim, X.-Z. Wang, Q.M.J. Wu, A review of generalized zero-shot learning methods, IEEE Trans. Pattern Anal. Mach. Intell. 45 (4) (2023) 4051–4070, http://dx.doi.org/10.1109/TPAMI.2022.3191696.

[25] A. Gupta, S. Narayan, S.H. Khan, F.S. Khan, L. Shao, J. van de Weijer, Generative multi-label zero-shot learning, 2021, CoRR arXiv:2101.11606, URL https://arxiv.org/abs/2101.11606.

[26] T. Xu, Y. Zhao, X. Liu, Dual generative network with discriminative information for generalized zero-shot learning, Complexity 2021 (2021) 6656797, http://dx.doi.org/10.1155/2021/6656797.

[27] G.-S. Xie, L. Liu, X. Jin, F. Zhu, Z. Zhang, J. Qin, Y. Yao, L. Shao, Attentive region embedding network for zero-shot learning, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 9376–9385, http://dx.doi.org/10.1109/CVPR.2019.00961.

[28] J. Ba, K. Swersky, S. Fidler, R. Salakhutdinov, Predicting deep zero-shot convolutional neural networks using textual descriptions, 2015, http://dx.doi.org/10.48550/ARXIV.1506.00511, URL https://arxiv.org/abs/1506.00511.

[29] D. Wang, Y. Li, Y. Lin, Y. Zhuang, Relational knowledge transfer for zero-shot learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 30, (1) 2016, http://dx.doi.org/10.1609/aaai.v30i1.10195, URL https://ojs.aaai.org/index.php/AAAI/article/view/10195.

[30] Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, B. Schiele, Latent embeddings for zero-shot classification, 2016, http://dx.doi.org/10.48550/ARXIV.1603.08895, URL https://arxiv.org/abs/1603.08895.

[31] P. Morgado, N. Vasconcelos, Semantically consistent regularization for zero-shot recognition, 2017, http://dx.doi.org/10.48550/ARXIV.1704.03039, URL https://arxiv.org/abs/1704.03039.

[32] S. Mannor, D. Peleg, R. Rubinstein, The cross entropy method for classification, in: Proceedings of the 22nd International Conference on Machine Learning, 2005, pp. 561–568.

[33] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, http://dx.doi.org/10.48550/ARXIV.1412.6980, URL https://arxiv.org/abs/1412.6980.

[34] A.S.M.M. Hasan, F. Sohel, D. Diepeveen, H. Laga, M.G.K. Jones, A survey of deep learning techniques for weed detection from images, 2021, http://dx.doi.org/10.48550/ARXIV.2103.01415, URL https://arxiv.org/abs/2103.01415.

[35] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U.R. Acharya, V. Makarenkov, S. Nahavandi, A review of uncertainty quantification in deep learning: Techniques, applications and challenges, Inf. Fusion 76 (2021) 243–297, http://dx.doi.org/10.1016/j.inffus.2021.05.008, URL https://www.sciencedirect.com/science/article/pii/S1566253521001081.

[36] Queensland Government, Restrictive invasixe plants, 2021, URL https://www.business.qld.gov.au/industries/farms-fishing-forestry/agriculture/land-management/health-pests-weeds-diseases/weeds-diseases/invasive-plants/restricted.

[37] J. Pennington, R. Socher, C.D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, 2014, pp. 1532–1543.

[38] P. Morgado, N. Vasconcelos, Semantically consistent regularization for zero-shot recognition, 2017, CoRR arXiv:1704.03039, URL http://arxiv.org/abs/1704.03039.

[39] D. Wang, Y. Li, Y. Lin, Y. Zhuang, Relational knowledge transfer for zero-shot learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 30, (1) 2016, http://dx.doi.org/10.1609/aaai.v30i1.10195, URL https://ojs.aaai.org/index.php/AAAI/article/view/10195.

[40] N. Tishby, N. Zaslavsky, Deep learning and the information bottleneck principle, 2015, http://dx.doi.org/10.48550/ARXIV.1503.02406, URL https://arxiv.org/abs/1503.02406.

[41] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009, pp. 248–255.

[42] Z. Han, Z. Fu, S. Chen, J. Yang, Contrastive embedding for generalized zero-shot learning, 2021, http://dx.doi.org/10.48550/ARXIV.2103.16173, URL https://arxiv.org/abs/2103.16173.

[43] Z. Chen, S. Wang, J. Li, Z. Huang, Rethinking generative zero-shot learning: An ensemble learning perspective for recognising visual patches, in: Proceedings of the 28th ACM International Conference on Multimedia, Association for Computing Machinery, New York, NY, USA, 2020, pp. 3413–3421, http://dx.doi.org/10.1145/3394171.3413813.