

# Deep-CNN based Robotic Multi-Class Under-Canopy Weed Control in Precision Farming

Yayun Du<sup>1</sup>, Guofeng Zhang<sup>1</sup>, Darren Tsang<sup>1</sup>, Mohammad Khalid Jawed<sup>1</sup>

**Abstract**—Smart weeding systems to perform plant-specific operations can contribute to the sustainability of agriculture and the environment. Despite monumental advances in autonomous robotic technologies for precision weed management in recent years, work on under-canopy weeding in fields is yet to be realized. A prerequisite of such systems is reliable detection and classification of weeds to avoid mistakenly spraying and, thus, damaging the surrounding plants. Real-time multi-class weed identification enables species-specific treatment of weeds and significantly reduces the amount of herbicide use. Here, our first contribution is the first adequately large realistic image dataset *AIWeeds* (one/multiple kinds of weeds in one image), a library of about 10,000 annotated images of flax and the 14 most common weeds in fields and gardens taken from 20 different locations in North Dakota, California, and Central China. Second, we provide a full pipeline from model training with maximum efficiency to deploying the TensorRT-optimized model onto a single board computer. Based on *AIWeeds* and the pipeline, we present a baseline for classification performance using five benchmark CNN models. Among them, MobileNetV2, with both the shortest inference time and lowest memory consumption, is the qualified candidate for real-time applications. Finally, we deploy MobileNetV2 onto our own compact autonomous robot *SAMBot* for real-time weed detection. The 90% test accuracy realized in previously unseen scenes in flax fields (with a row spacing of 0.2-0.3 m), with crops and weeds, distortion, blur, and shadows, is a milestone towards precision weed control in the real world. We have publicly released the dataset and code to generate the results at <https://github.com/StructuresComp/Multi-class-Weed-Classification>.

## I. INTRODUCTION

Herbicides can have negative side-effects on the ecosystem, biodiversity, and human health [1]. Conventional weed control methods indiscriminately spray the entire field, including soil, crops, and weeds, with a single herbicide. This strategy is widely applied as it does not require the users to know the spatial distribution or the type of the weeds. However, overuse of chemicals has led to hundreds of herbicide-resistant species of weed across the world [2]. Reducing the amount of herbicides is a crucial step towards sustainable agriculture. Site-specific weed control can result in 90% savings in herbicide expenditures [3]. Since the worldwide annual sales of pesticides are on the order of a hundred billion dollars [4], the economic impact – in addition to environmental benefits – of precision weed management is overwhelming.

The past decade has seen revolutionary advances in mobile robotic platforms for precision agriculture and several commercial entities (Blue River Technology, ecoRobotix,

Hitch Robotics, EarthSense, and Naio Technologies) have commercialized various robotic vehicles. Real-time weed control is the primary objective of several of these vehicles, e.g. “See and Spray Machines” of Blue River Technology and TerraSentia of EarthSense. In the context of these ongoing activities in commercial as well as academic [5] sectors, real-time weed detection and classification is a truly enabling technology for robotics. However, three critical challenges – (1) data, (2) training, and (3) real-world deployment – have to be overcome for fully autonomous weed management. While many weed libraries have been released [6], [7], [8], [9], [10], what is lacking is a large enough image dataset of realistic fields, including crops and weeds, distortion, shadows, and motion blur. The onboard computing power of agricultural robots can often be limited, especially for row crops (e.g. flaxseed and canola) where the inter-row spacing is small as 0.2-0.3 m. This spacing limits the size of the robot. Existing platforms except for robots from ecoRobotix are suitable only for fields with wide row spacing ( $\geq 0.4$  m) as they use GPS (limited accuracy with crop blocking and size-incompatible with small robots) for navigation and guidance. Moreover, the robot should be able to travel under-canopy once the crop canopy has been established for truly precise weed management. This further restricts the size of the robot as well as the computing capability. Small vehicles to travel under-canopy have been introduced by EarthSense. However, owing to the challenges above, autonomous weed management with small vehicles is yet to be achieved.

To address these issues, we employ our own low-cost (around \$500) compact robot, *SAMBot*, as shown in Fig. 1(a1), to which two cameras are attached 0.2 – 0.4m above the ground. The customized three-degree-of-freedom gimbal enables the camera to scan an angle between 0° to 150°. We employ a mobile Convolutional Neural Network (CNN) for weed classification. We chose this CNN because it is the next-generation of on-device computer vision networks, can predict much faster than other networks, and maintain competitive performance. The network utilizes a modified MobileNetV2 [11] architecture (i.e. dropping last two fully-connected layers and adding one global average pooling layer) and its inference is optimized by NVIDIA TensorRT. Using the same pipeline, we also train and run another 4 models with acceptable training speed and accuracy. The contributions of this system paper are as follows:

- We release a realistic dataset – *AIWeeds* – containing 10,000 labeled images with multiple weeds/crops in view, including flax and 14 most common weeds in

<sup>1</sup>Department of Mechanical & Aerospace Engineering, University of California, Los Angeles, 420 Westwood Plaza, Los Angeles, CA 90095

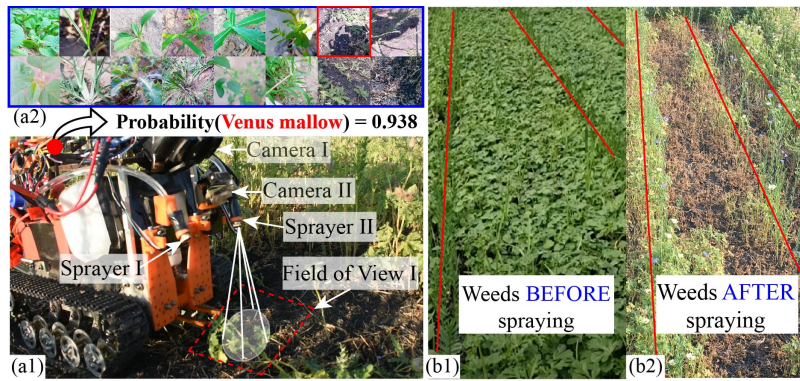


Fig. 1: (a1) The cameras on our *SAMBot* robot performing multi-weed classification to recognize the weed(s) in the view of camera I (shown as the dashed red rectangle) in the flax fields by exploiting the light-weight MobileNetV2 CNN model and executing spraying (continuous white lines); (a2) The predicted weed – Venus mallow (in the small red rectangle) in the view of camera I out of the 16-class *AIWeeds* (in the blue rectangle) by onboard Jetson Nano (denoted as the filled red dot in (a1)); Snapshots showing the statuses of weeds (b1) before and (b2) after robotic herbicide spraying. Continuous lines in both (b1) and (b2) represent the flax croplines. Note that weed classification is for point spray but not for this dense weed scenario, so (b1) and (b2) are here to validate our robotic spraying system.

North Dakota, the middle of China, and California [12]. The images are taken at several stages of weed from 20 different sites in the fields and gardens with variations in illumination, shadows, weather conditions, view perspective and distance, and plant growth stages.

- We demonstrate a multi-class weed classification pipeline for model deployment on GPU boards, even low-end boards such as Jetson Nano/Xavier Nx. Using this pipeline, we study weed classification using 5 modified models: CNN-MobileNetV2 and 4 other more complicated models and compare their performance through various metrics. Researchers can use this study to evaluate their unique situational needs, such as training time and accuracy, and select the appropriate model for their needs.
- We run CNN-MobileNetV2, the only qualified candidate for onboard under-canopy weed classification, on *SAMBot* (with Jetson Nano) in flax fields and achieve satisfactory experimental results, i.e. accurately detect and spray weeds. This is a milestone for real-time autonomous weeding in realistic fields.

The remainder of this study is arranged as follows. Section II presents the state-of-the-art on weed classification. Section III describes how the dataset is built while Section IV gives details of our multi-class weed classification pipeline. We then show our experimental results of models trained on our dataset and deployed onto the real-time robot system. Finally, the conclusion is drawn in Section VI.

## II. RELATED WORK

### A. Vision-Based Weed Control & Datasets

Image-based [13], [14], [6], spectrum-based [15], [16] and spectral image-based [17], [18] methods have been successfully applied to identify weeds from both ground and aerial photography. Spectrum and spectral image-based approaches

are ideally suited for highly controlled site-specific environments where spectrometers can be tailored for consistent acquisition and detection. Nonetheless, it is challenging to incorporate them in harsh field environment and deploy them onto compact vehicles. Vision-based methods, on the other hand, benefit from cheaper and simpler image acquisition under varying illumination conditions. Weed datasets in realistic fields are indispensable for vision-based weed control, but the existing ones [6], [7], [8], [9] only embrace ideal weed images without background, none of which is apt for real field applications. The most realistic one [10] consists of static clear (without motion blur and shadows) images shot from a straight downwards view. Nonetheless, distortion, shadows, and motion blur that always appear during applications increasing technical difficulty are not reflected in [10].

### B. Multi-class Classification Using Deep Neural Network

Classical vision-based weed classification methods rely on different features of crop plants and weeds, such as color, leaf shape [19] and size, vein patterns, and so forth [20]. However, in complex natural scenarios with high weed densities where weeds and crop plants overlap and occlude, they cannot perform the task correctly and robustly. In addition, further investigation is needed as to whether they are applicable to actual field conditions. This problem is addressed by recent deep learning models, such as Convolutional Neural Networks (CNNs). CNNs [8], [21], [22], [23], [24] take advantage of a deep hierarchical structure to extract global features of the image and context information (background such as soil), which significantly reduces the error rate of image recognition than the classical algorithms mentioned above. For early-stage wide weed control (around stem elongation but before booting), many current methods for weed detection and classification focus on segmenting images with both RGB and excess green/red or near-infrared [5], [21],

[25] recordings. However, they are powerless in fields with canopies developed and visual occlusion. Performing early-stage weed control will probably destroy or affect the growth of crops and thus the crop yield. As such, under-canopy weed control is a must. [21] and [7] tried semantic segmentation of weeds and crops. Still, segmentation methods [21], [7], [21], [25] require lots of effort and time because of hand-crafted labeling and are challenging to be run on an embedded computer. As a result, there is a timely need to do under-canopy multi-class weed identification in real fields with CNNs.

### C. Under-canopy Weeding with Low-cost Robots

As mentioned above, CNN models have been widely explored by researchers and run on expensive, powerful PCs [21], [7], [25], [8]. However, none of the prior works deployed their model onto a flexible mobile robot that works for under-canopy weeding. Powerful PCs are usually not an option for small economical platforms. Our prior work verified that our low-cost robot platform *SAMBot* worked autonomously in real fields with narrow row spacing, e.g., flax and canola fields. On *SAMBot*, NVIDIA Jetson Nano is used as a cost effective (\$99) solution that possesses the performance and capabilities to run modern artificial intelligence workloads.

In summary, to the best of our knowledge, this work is the first trial applying lightweight multi-class models to agricultural robots for under-canopy weed control and testing the performance in real flax fields.

## III. DATASET COLLECTION

Our first goal is to create a variable and realistic dataset that allows us to step towards the further objective, i.e. to realize and enhance the baseline accuracy of the off-the-shelf CNNs and make it easy to be trained and deployed to facilitate wider use of the dataset. Finally, the deployed model enables *SAMBot* to detect, identify, and precisely spray weeds during field marching, even when the visual appearance of the plants and background has changed. In this section, we provide details of how we collected images to reflect varying scenes and target variability in realistic fields.

Unlike the aforementioned libraries [6], [7], [8], [9], [10], we build a dataset named *AIWeeds* containing flax and 14 most common weeds in the fields in North Dakota, California, and central China. The image resolution is  $1920 \times 1080$  or  $1280 \times 720$  pixels. Table I shows the weed species and their corresponding quantity in *AIWeeds*. Full and corresponding abbreviated names of the weeds are: *Amaranthus spinosus* (AS.), *Brachypodium sylvaticum* (BS.), *Cirsium arvense* (CA.), *Cynodon dactylon* (CD.), *Dandelion* (D.), *Lamb-squarters* (L.), *Nutsedge* (N.), *Plantago Major* (PM.), *Setaria faberi* (SF.), *Sonchus arvensis* (SA.), *Verdolagas Pirsplane* (VP.), *Venus mallow* (VM.), *Canada thistle* (CT.), *Flax*, and *Negatives* (Neg.). A total of more than 10,000 images were taken under different sunlight (from 7 a.m. to 6 p.m.), weather conditions (super bright, sunny, cloudy, and rainy), growth stages (from sprouting to ripening), and varying

health conditions (under drought, plant disease, and insect pest infection). On average, 600 images of each target species were taken from at least three different locations. Rotation and scale of the target weed species in the images also vary as they are photographed *in situ* with unknown orientation. Fig. 2 displays some image samples from our dataset and gives a sense of the variations. These variations in conditions were deliberate in order to significantly increase the generality of *AIWeeds*. They can, however, influence the foliage color, strength of features, and other noticeable anomalies. This includes the intra-species variation of the data. As shown in Fig. 2, another variability in our dataset arises from the complex and dynamic target backgrounds. Although our dataset is large, over-fitting might still appear, the main concern of deep neural networks. Later in Section IV, we will describe the image augmentation skills implemented to avoid over-fitting. We also kept this in mind while constructing *AIWeeds*. We rotated the camera and rendered motion blur during the shot. These confounding factors of heterogeneity in *AIWeeds* will jointly lead to deeper and more complex models to attain acceptable performance.

In addition, locations subject to dense weed infestations are also populated by other native plants. Since we are unable to process a dataset including all plants, all other non-target species in view must be labeled as *negative* samples, along with all non-target background images. Unfortunately, this introduces a highly variable class in the dataset that will be difficult to classify consistently. In order to prevent over-fitting, increase the accuracy and robustness of weed classification models despite the disturbance of non-target plants, we include the *negative* class in *AIWeeds* (abbreviated as Neg.) in Table I.

The strictness of the collection process will ensure the accuracy and robustness of all classification learning models. Fig. 2 displays a subset of *AIWeeds* to demonstrate its variety and generality within classes, from which the complexity of the learning problem is evident.

## IV. MULTI-CLASS WEED CLASSIFICATION PIPELINE

In this section, we illustrate the pipeline for a multi-class weed classification scheme. To implement and realize our models, we utilized the popular machine learning framework, Tensorflow, and high-level API, Keras. This allowed us to try out various models, such as VGG19, Nasenet-Mobile, ResNet50, InceptionV3, Xception, DenseNet, and MobileNetV2. After testing different models, we concluded that MobileNetV2 provided the best combination of low memory usage and computational time but maintained a respectable level of accuracy. It is, therefore, deployable onto Jetson Nano with limited onboard computation resources.

All models were pretrained to recognize 1000 object classes in ImageNet, and we slightly modified their architectures to classify 16 classes (14 types of weeds, flax, and negatives) in *AIWeeds*. Modifications will be illustrated on MobileNetV2 because they are identical for other models. As shown in Fig. 3, the last fully connected layer consisting of 1,000 neurons of ImageNet-trained MobileNetV2 is replaced

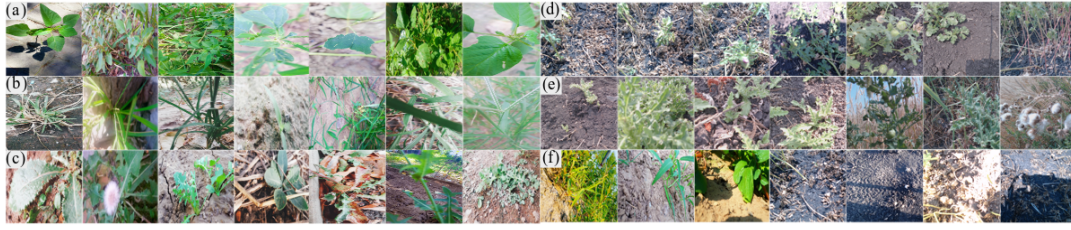


Fig. 2: Sample images from some classes of the *AIWeeds* dataset to show the variation: (a) *Amaranthus spinosus* (AS.), (b) *Brachypodium sylvaticum* (BS.), (c) *Cirsium arvense* (CA.), (d) Venus mallow (VM.), (e) Canada thistle (CT.), (f) Negatives (NG.).

TABLE I: The weed species collected in our dataset and their corresponding quantity.

| Weed        | AS. | BS. | CT. | CA. | CD. | D.  | Flax | L.  | Neg. | N.  | PM. | SF. | SA. | VM. | VP. |
|-------------|-----|-----|-----|-----|-----|-----|------|-----|------|-----|-----|-----|-----|-----|-----|
| # of Images | 659 | 655 | 560 | 990 | 631 | 428 | 625  | 549 | 1474 | 649 | 526 | 566 | 565 | 704 | 559 |

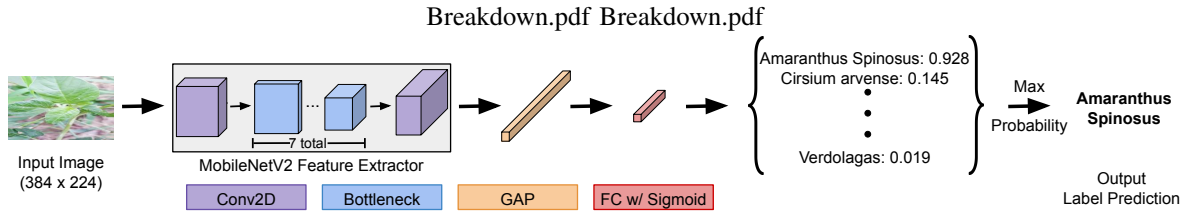


Fig. 3: Detailed architecture of learning model for multi-weed classification with modifications on MobileNetV2.

by a 16-neuron fully connected layer. We used MobileNetV2 as a feature extractor and added two layers at the end: a global average pooling (GAP) layer and a fully connected layer that used sigmoid as the activation function. Sigmoid is chosen because the probability of every class presenting in the same view (image) in nature is the same. It allows the output layer to identify the likelihood of an image belonging to each class. The weed with the highest sigmoid-activated neuron probability is thought to appear in the input image.

The preprocessing preparation for learning includes image flipping, resizing, and augmentation. First, all images for training, validation, and testing were rotated and resized to  $384 \times 224$ , the size closest to the default  $224 \times 224$  in ImageNet while matching the normal output of the robot's camera and keeping the ratio of our photographs taken to prevent excessive distortion. Next, image augmentation was performed by rotating every image arbitrarily in the range of  $[-360^\circ, 360^\circ]$  and then scaling it in the range of  $[0.5, 1]$  both horizontally and vertically. After that, each color channel and pixel intensity were both randomly shifted between -25 pixels to 25 pixels to account for the illumination variance. We also randomly scaled pixel intensity within  $[0.75, 1.25]$  range and did random perspective transformations on each image to stimulate a wide range of viewing distances and angles. In summary, our image augmentation implementations accounted for variations in rotation, scale, illumination, color, and perspective. Otherwise, the deep neural network models mentioned above, with trainable weights in the order of millions, would dramatically over-fit the images by memorizing the training subsets.

Then, we split the labeled images in *AIWeeds* into training,

validation, and testing sets; these sets contained 60-20-20 percent respectively for  $k$ -fold cross validation with  $k = 5$ . Stratified random partitioning was executed to ensure even distribution of different weed classes within each subset. A random split of 60% formed the training dataset, while 20% constituted the validation dataset to monitor the training process and minimize over-fitting. The remaining 20% were reserved for testing and never allowed to join any training procedure. Normally, training the models from scratch with our custom dataset cannot guarantee acceptable performance even after a long training time on a computationally powerful platform. As a result, each model was loaded with its corresponding pre-trained weights on ImageNet as initial weights before training through Keras. The weights of the fully-connected layer were initialized by uniform distribution.

Finally, we fine-tuned the layers using our custom built dataset. The standard binary cross-entropy loss function and Adam optimizer were used to train all models. Batches of 32 images were produced for training, which would be aborted if the validation loss did not decrease after 32 epochs. Here, the validation loss refers to the classification error calculated on the validation subset of images. The training was restarted after an abortion by loading the continuously saved model with the smallest running validation loss. After exploration, the initial learning rate was set as 0.0001 and was then successively halved every time the validation loss did not decrease after 16 epochs. The learning rate would be reduced to  $0.5 \times 10^{-4}$  when the training restarted after an abortion. The validation and testing results of all models will be given in Section V-B.

Experiments demonstrated that only MobileNetV2 could

be deployed and run on hardware-limited *SAMBot* (with Jetson Nano) with/without structure optimization by TensorRT, while other complicated models are not deployable even with TensorRT speedup. We used TensorRT to optimize the inference time that delivers low latency, memory usage, and high throughput. The four key operation steps related to TensorRT include creating frozen graphs for trained models, converting frozen graph to the TensorRT engine, running TensorRT engine, and benchmarking all models. Details are given in our open-source repository.

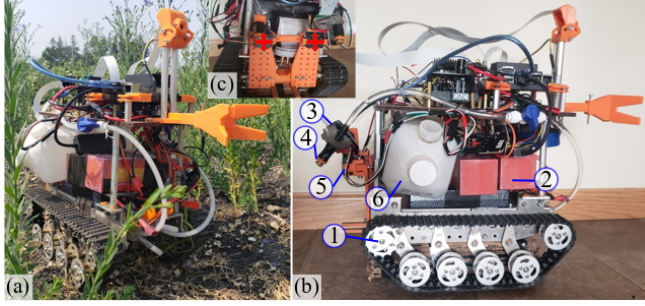


Fig. 4: (a) *SAMBot* working in flax fields; (b) The side-view of the hardware layout of *SAMBot*; (c) The back-view of the adjustable “see and spray” system on *SAMBot*.

## V. EXPERIMENTS & RESULTS

In this section, we illustrate our experimental setup, followed by the quantitative assessment of five 16-class classification approaches. Then, we deploy MobileNetV2, onto our embedded board, Jetson Nano (the red dot in Fig. 1(a1)). Finally, we run our robot, in flax fields in North Dakota, with real-time video streaming. This is to verify the capabilities of the deployed model, i.e. whether the robot can successfully detect multiple types of weeds in flax fields with a medium weed density and spray the corresponding herbicide. The experimental results and supplementary video validate the practicality of MobileNetV2 on *SAMBot*.

### A. Experimental Setup

All experiments were conducted on our miniaturized, low-cost, functional agricultural robot – *SAMBot* – in Fig. 1(a1) and Fig. 4. It is developed and tested for weed control in flax (as shown in Fig. 4(a1)) and canola fields of North Dakota, the leading producer with 91% of the U.S. flax production and 85% of canola production [26]. The robot is generally applicable to row crops. The robot has a powerful drive train (1) (the rated torque of motors is 4.81N·m). It successfully passed all the bumps/dents (the maximum height/depth of which is the same as the height of chassis) and finished the full exploration of the fields during field tests during the Summer of 2021 in Fargo, North Dakota. It continuously worked in the fields more than 14 hours with 12-cell 18400mAh onboard LiPo batteries (2). Referring to Fig. 1(a1) and Fig. 4(b), two cameras (3) are mounted at the back of the robot, about 20-40 cm above the ground. The images in *AIWeeds* were therefore taken from the robot's

perspective. One camera and one pressure sprayer (4) are rigidly coupled and actuated together by a gimbal (5) to realize a yaw angle of  $0^\circ - 150^\circ$ . Each sprayer is connected to a herbicide tank (6). Not only the orientation of the gimbal assemblies can be changed by servos, but also their positions are adjustable left and right, up and down on the 3D-printed pegboard based on the growth stage of crops/weeds and row spacing, as indicated in the red arrows in Fig. 4(c).

### B. Quantitative Results of Workstation Training

We use the  $F_1$  score for quantitative evaluation:

$$F_1(s) = 2 \cdot \frac{\text{precision}_s \cdot \text{recall}_s}{\text{precision}_s + \text{recall}_s}, \quad (1)$$

where  $\text{precision}_s$ ,  $\text{recall}_s$  are the precision, recall for class  $s$ , respectively. All models presented in this section are trained and tested with the dataset in Table I.

Fig. 5 shows the training loss and average validation class accuracy of models over 64 epochs on our *AIWeeds* dataset. On our GTX 1080 Ti platform, each epoch of DenseNet121, InceptionV3, MobileNetV2, ResNet50, and Xception took 978s, 437s, 274s, 672s, and 672s on average, respectively. **Training loss and average validation class accuracy** of different models are plotted in Fig. 5 to eliminate overfitting during training. MobileNetV2 offers an accuracy of 94.50%, while being the fastest to finish training, far ahead of other models. DenseNet121, on the other hand, gives the best accuracy, 96.77%, and the smallest loss, 0.0027, after being trained for 64 epochs. We continued to train MobileNetV2 for 128 epochs, which doubled the time needed for 64 epochs, and give a second-best accuracy of 96.15%. The speed at which the algorithms above run is proportional to the total number of parameters in the network that are available via the Internet. We observe that MobileNetV2 takes 1/5-1/4 the time to train as other more complicated algorithms.

Fig. 6 displays the  $F_1$ -score of the five trained models on each kind of weed. All models perform well (above 90% for

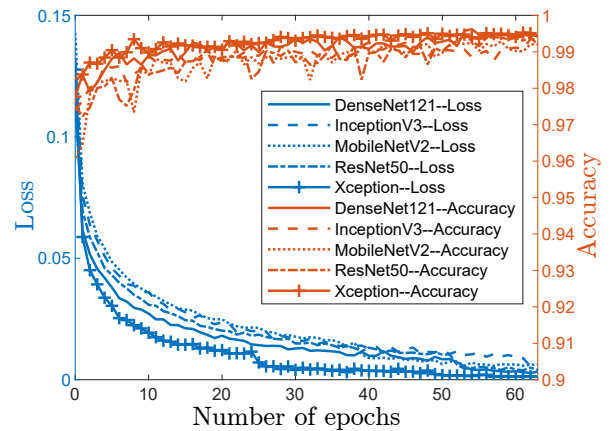


Fig. 5: **Training loss and average validation class accuracy** of different models. The maximum number of epochs is set to 64 for plotting but in reality, MobileNetV2 was trained for 128 epochs.

all classes) considering the complexity of *AIWeeds* as mentioned in Section III. The  $F_1$ -score of Neg. (backgrounds) class is relatively low. Looking carefully into the confusion matrix of all models, we find that most mispredictions are between Neg. and other weeds. This makes sense as the number of Neg. in *AIWeeds* is double as other weeds, as shown in Table I. The targeted plant takes a smaller area out of the overall image. When we took pictures of it, we did not avoid other native non-targeted plants in view, so it would be more difficult to be identified from a noisy background. One surprising finding is that BS. is narrow-leaf and looks similar to SF. from human recognition, especially at an early stage and in a noisy background, but were distinguished well by CNN models. This verifies the ability of CNNs to use a deep hierarchical structure to extract features.

### C. Flax Field Experiments

All of the above well-trained models besides MobileNetV2 cannot be deployed onto *SAMBot* (with Jetson Nano). Considering the running speed of the robot, the resolution of the two cameras as shown in Figs. 1(a1) and 4 is set to  $384 \times 224$ , the same as the input of CNN models. The frame rate is set to 10 frames per second. Experimentally, with MobileNetV2 running, we ran *SAMBot* in flax fields in North Dakota with medium weed density (with VM. and CT.) for 15m, and the robot was able to classify each weed at an average accuracy of 90%. Meanwhile, it consumed less herbicide than commercial sprayers for the same spraying range (the flux rate of our customized pressurized sprayer is 78ml/min while 95.6ml/min for a commercial two-sprayer system). The status of weeds before and after herbicide spraying is shown in Figs. 1(b1) and (b2), respectively. Fig. 7 shows the field test results. The robot successfully recognized and sprayed

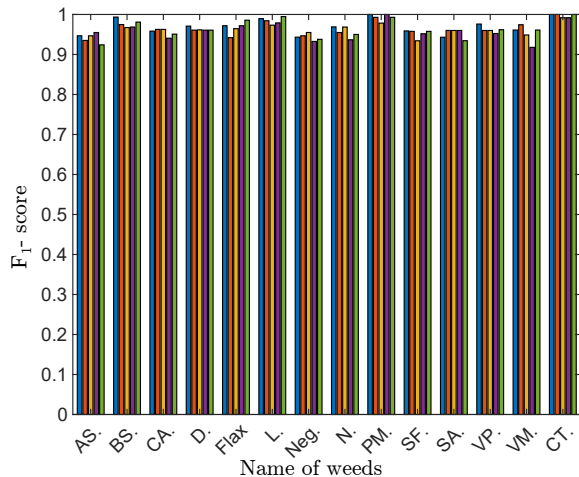


Fig. 6:  $F_1$ -score of 5 models per weed class (horizontal axis). The 5 models at each class from left to right are DenseNet121 (blue, 64 epochs), InceptionV3 (brown, 64 epochs), MobileNetV2 (yellow, 128 epochs), ResNet50 (purple, 64 epochs), and Xception (green, 64 epochs), respectively.



Fig. 7: Images extracted from the video streaming on the robot. The prediction results of MobileNetV2 (128 epochs) model are (a) VM. (correct), (b) VM. (correct), (c) flax (correct), (d) CT. (correct), (e) Neg. (correct), and (f) CT. (wrong; ground truth is VM.).

VM. in (a) and (b) though there was an apparent disturbance from CT. (in rectangles). It also succeeded in detecting flax in (c), spraying CT. in (d) under VM.'s appearance. The 10% failures mainly consist of cases where the leaves of VM. looked similar to CT. under strong light exposure as displayed in (f).

### VI. CONCLUSIONS

In summary, we introduce the first large, realistic multiclass weed image dataset, *AIWeeds*, with considerable variation (e.g. shadows, lighting and perspective change, different plant growth stages, and so on), collected entirely from *in situ* in flax fields or gardens. It consists of flax, the 14 most common weeds, and backgrounds collected from North Dakota and California (U.S.) and middle China. Based on our dataset, we present baseline performance of five benchmark CNN models – DenseNet121, InceptionV3, MobileNetV2, ResNet50, and Xception – all of which perform well with an average  $F_1$ -score above 90% on a highly variable dataset. Our low-cost, compact *SAMBot* with a computational resource-limited onboard Jetson Nano is then run in flax fields with a medium density of Venus mallow and Canada thistle after the lightest MobileNetV2 being deployed. It realized a weed classification accuracy of 90% with an inference time of 47.78ms. Evaluations on both *AIWeeds* dataset and experiments in real fields demonstrate that our system is (i) applicable to under-canopy weeding, (ii) adaptable to unseen scenes, and (iii) able to robustly classify weeds at varying growth stages and environments. Our work (including a dataset, multi-class weed classification pipeline, and experimental results) is a milestone towards under-canopy weed control in fields. The data and tools introduced in this paper enable various commercial mobile robots to detect, classify, and manage weeds of multiple types and thus reduce the amount of herbicide use by at least an order of magnitude.

### ACKNOWLEDGMENTS

We acknowledge support from the NSF (Award # IIS - 1925360) and the National Institute of Food and Agriculture, USDA (Award # 2021-67022-34200). Special thanks to Afrina Rahman and Prof. Mukhlesur Rahman from North Dakota State University for their field preparation and herbicide spraying instructions, Mr. Tongqing Du for helping

build *AIWeeds*, and Dr. Naiyu Gao at the Chinese Academy of Sciences for algorithm discussion.

## REFERENCES

- [1] L. Horrigan, R. S. Lawrence, and P. Walker, "How sustainable agriculture can address the environmental and human health harms of industrial agriculture,," *Environmental health perspectives*, vol. 110, no. 5, pp. 445–456, 2002.
- [2] H. M. LeBaron and J. McFarland, "Herbicide resistance in weeds and crops: an overview and prognosis," 1990.
- [3] C. Timmermann, R. Gerhards, and W. Kühbauch, "The economic impact of site-specific weed control," *Precision Agriculture*, vol. 4, no. 3, pp. 249–260, 2003.
- [4] F. P. Carvalho, "Pesticides, environment, and food safety," *Food and Energy Security*, vol. 6, no. 2, pp. 48–60, 2017.
- [5] P. Lottes, J. Behley, N. Chebrolu, A. Milioto, and C. Stachniss, "Joint stem detection and crop-weed classification for plant-specific treatment in precision farming," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8233–8238, IEEE, 2018.
- [6] A. dos Santos Ferreira, D. M. Freitas, G. G. da Silva, H. Pistori, and M. T. Folhes, "Weed detection in soybean crops using convnets," *Computers and Electronics in Agriculture*, vol. 143, pp. 314–324, 2017.
- [7] M. Dyrmann, H. Karstoft, and H. S. Midtiby, "Plant species classification using deep convolutional neural network," *Biosystems Engineering*, vol. 151, pp. 72–80, 2016.
- [8] 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," *Scientific reports*, vol. 9, no. 1, pp. 1–12, 2019.
- [9] S. H. Lee, C. S. Chan, P. Wilkin, and P. Remagnino, "Deep-plant: Plant identification with convolutional neural networks," in *2015 IEEE international conference on image processing (ICIP)*, pp. 452–456, IEEE, 2015.
- [10] N. Chebrolu, P. Lottes, A. Schaefer, W. Winterhalter, W. Burgard, and C. Stachniss, "Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1045–1052, 2017.
- [11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.
- [12] Database for weed classification, Structures-Computer Interaction Laboratory, University of California, Los Angeles. <https://github.com/StructuresComp/Multi-class-Weed-Classification/>.
- [13] S. H. Lee, C. S. Chan, S. J. Mayo, and P. Remagnino, "How deep learning extracts and learns leaf features for plant classification," *Pattern Recognition*, vol. 71, pp. 1–13, 2017.
- [14] A. Bakhshipour and A. Jafari, "Evaluation of support vector machine and artificial neural networks in weed detection using shape features," *Computers and Electronics in Agriculture*, vol. 145, pp. 153–160, 2018.
- [15] A. Shirzadifar, S. Bajwa, S. A. Mireei, K. Howatt, and J. Nowatzki, "Weed species discrimination based on simca analysis of plant canopy spectral data," *Biosystems Engineering*, vol. 171, pp. 143–154, 2018.
- [16] L. Li, X. Wei, H. Mao, and S. Wu, "Design and application of spectrum sensor for weed detection used in winter rape field," *Transactions of the Chinese Society of Agricultural Engineering*, vol. 33, no. 18, pp. 127–133, 2017.
- [17] M. Louargant, G. Jones, R. Faroux, J.-N. Paoli, T. Maillot, C. Gée, and S. Villette, "Unsupervised classification algorithm for early weed detection in row-crops by combining spatial and spectral information," *Remote Sensing*, vol. 10, no. 5, p. 761, 2018.
- [18] F. Lin, D. Zhang, Y. Huang, X. Wang, and X. Chen, "Detection of corn and weed species by the combination of spectral, shape and textural features," *Sustainability*, vol. 9, no. 8, p. 1335, 2017.
- [19] T. Beghin, J. S. Cope, P. Remagnino, and S. Barman, "Shape and texture based plant leaf classification," in *International Conference on Advanced Concepts for Intelligent Vision Systems*, pp. 345–353, Springer, 2010.
- [20] N. Li, X. Zhang, C. Zhang, L. Ge, Y. He, and X. Wu, "Review of machine-vision-based plant detection technologies for robotic weeding," in *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 2370–2377, IEEE, 2019.
- [21] P. Lottes, M. Hoferlin, S. Sander, M. Müter, P. Schulze, and L. C. Stachniss, "An effective classification system for separating sugar beets and weeds for precision farming applications," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5157–5163, IEEE, 2016.
- [22] I. Sa, Z. Chen, M. Popović, R. Khanna, F. Liebisch, J. Nieto, and R. Siegwart, "weednet: Dense semantic weed classification using multispectral images and mav for smart farming," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 588–595, 2017.
- [23] A. Milioto, P. Lottes, and C. Stachniss, "Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns," in *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 2229–2235, IEEE, 2018.
- [24] M. Fawakherji, A. Youssef, D. Bloisi, A. Pretto, and D. Nardi, "Crop and weeds classification for precision agriculture using context-independent pixel-wise segmentation," in *2019 Third IEEE International Conference on Robotic Computing (IRC)*, pp. 146–152, IEEE, 2019.
- [25] P. Lottes, J. Behley, A. Milioto, and C. Stachniss, "Fully convolutional networks with sequential information for robust crop and weed detection in precision farming," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2870–2877, 2018.
- [26] Four-year average from 2014–2017. United States Department of Agriculture. <https://www.nass.usda.gov/>.