DeepWeeds: A Multiclass Weed Species Image Dataset for Deep Learning - A Team Study on CNN

Aniruddha Anand Damle[1] Prakriti Biswas[1] Aditya Shrikant Kaduskar[1]

[1]Ira A. Fulton Schools of Engineering, Arizona State University

I. What problem was solved? Why is it significant?

Robotic weed management offers a significant increase in agricultural output.[1], [2] The key advantages of autonomous weed management systems are labor cost savings and the possibility for reduced herbicide consumption due to more effective delivery of herbicides to weed targets. Improving weed control effectiveness would have a huge economic impact. Farmers in Australia alone are expected to spend AUD $1.5 billion on weed management measures each year, with an additional $2.5 billion in lost agricultural production[3]. Successful agricultural robotics development is expected to decrease the aforementioned losses and increase output. The research paper that we have studied has focused on image-based techniques for the detection and classification of weed species, using deep learning models (ResNet-50 and Inception-v3).

II. How is raw data prepared and processed? Furthermore, what are the inputs and outputs of the deep networks?

The DeepWeeds dataset contains 17,509 labeled images from 8 weed species and one negative class, that have been divided into training, testing, and validation subsets in the ratio: 60:60:20, respectively. This dataset was divided for k-fold cross-validation with k = 5. Stratified random partitioning was employed to ensure a fair distribution of classes throughout each subset, except for the negative class, which is significantly larger. The training subset is made up of 60% of the images, while the validation subset consists of 20% of the images to monitor the training process and reduce overfitting. A random seed was used to regulate the random splits for each fold, allowing the individual split to be replicated as needed. The remaining 20% of the images were used for testing alone. To overcome overfitting, the training and validation image subsets were augmented to account for variations in rotation, scale, color, illumination, and perspective. Image augmentation was performed using OpenCV and its Python wrapper. All images were first resized to 256*256 pixels in size and randomly rotated in the range of [-360;+360] degrees. Then, each image was scaled along its width and height in the range of [0:5;1]. Pixel intensity was randomly shifted within the range of [-25,25] to account for illumination variance. Hence, all color channels were shifted uniformly. Pixel intensity was also randomly scaled within the range [0:75,1;2] for the same purpose. To imitate a wide range of viewing distances and perspectives, each picture was given a random perspective transform. As a final augmentation step, the images were flipped horizontally with a 50% probability and then cropped to the dimensions 224*224 pixels, as required for each architecture's input layer. The output of the layer. The output of the deep network is the 'DeepWeeds' dataset that comprises eight different weed species and various negative class plant life, that is native to Australia. This dataset comprises a total of 17,509 images.

III. Describe and introduce the deep network(s) used:

Transfer learning was used to train the models (Inception-v3[36] and ResNet-50[39]) to recognize the 1000 distinct ImageNet[38] object classes. A fully connected 9 neuron layer replaced the last fully connected 1,000 neuron layer. Each network was then given a b*r*c*3 input matrix, in which b is the number of images per training batch, r * c each denotes a spatial location in that image, and 3 is the quantity of image color channels. Both network outputs had the shape $b * r_a * c_a * f_a$ after removing the fully connected 1,000 neuron ImageNet output layer, in which a denotes the specific network and f_a denotes the number of extracted features for each spatial location $r_a * c_a$. The Inception-v3 spatial average pooling layer was used to convert the fully-convolutional $b * r_a * c_a * f_a$ output to the $b * f_a$ shape, which was then densely connected to the final b * 9 weed classification layer. The deployed global average pooling was nearly identical to the

Species	Black River	Charters Towers	Cluden	Douglas	Hervey Range	Kelso	McKinlay	Paluma	Total
Chinee apple	0	0	0	718	340	20	0	47	1125
Lantana	0	0	0	9	0	0	0	1055	1064
Parkinsonia	0	0	1031	0	0	0	0	0	1031
Parthenium	0	246	0	0	0	776	0	0	1022
Prickly acacia	0	0	132	1	0	0	929	0	1062
Rubber vine	0	188	1	815	0	5	0	0	1009
Siam weed	1072	0	0	0	0	0	0	2	1074
Snakeweed	10	0	0	928	1	34	0	43	1016
Negatives	1200	605	1234	2606	471	893	943	1154	9106
Total	2282	1039	2398	5077	812	1728	1872	2301	17509

Table 1. The distribution of DeepWeeds images by weed species (row) and location (column).

ResNet-50 model's 7*7 average pooling with 32 images per batch and 224*224*3 input images.

IV. What algorithm was used to train the deep networks?

Both models were trained using the Keras implementation of Adam[44], a first-order gradient-based stochastic optimization method for stochastic objective functions that are based on adaptive estimates of lower-order moments. The method is simple to implement, computationally efficient, requires little memory, is insensitive to gradient rescaling, and is well suited for problems with large amounts of data and/or parameters. The hyper-parameters have intuitive interpretations and do not require much tuning[44]. The starting learning rate (lr) was set to 1x10-4. It was then halved after every 16 epochs if the validation loss did not decrease. The training was done in 32-image batches and was stopped if the validation loss did not decrease after 32 epochs. The model with the smallest running validation loss was continuously saved during training to re-start the training after an abortion. In such cases, training was repeated with lr = 0.5x10-4 as the initial learning rate.

V. What results were shown, and what performance measures were used to show the effectiveness of the deep network(s)?

The DeepWeeds dataset was classified with the ResNet-50 and Inception-v3 CNN models to establish a baseline level of performance for future comparison. Results are tabulated in Table 2. In terms of accuracy and false positive rate, ResNet-50 surpassed Inception-v3. The ResNet-50 model's somewhat greater performance can be due to its slightly larger complexity, with 23.5 million trainable weights compared to 21.8 million for Inception-v3. The ResNet-50 model learns in a larger optimization space thanks to the added complexity. The precision statistics in Table 2 for the ResNet-50 model show that the model's positive predictions have a high level of confidence (greater than 90% for all species). Rubber vine, Parkinsonia, and Parthenium had the highest positive predictive values at 99.1%, 97.9%, and 96.7%, respectively. With an accuracy score of 96.%, the negative class prediction likewise has a high level of confidence. Chinee apple, lantana, thorny acacia, and snake plant, on the other hand, have the lowest relative confidence in their positive forecasts, ranging from 91.0 to 93.0 %. This might be owing to the high levels of misunderstanding between these specific weed species as a result of similar visual traits. Both models had an overall false-positive rate of roughly 2%, with individual false-positive rates for each weed species hovering around 1%. (as shown in Table 2). This significant finding suggests that when their models are deployed in the field, they will cause minimal misclassifications. The negative class, on the other hand, has a relatively high false-positive rate of 3.77% for Inception-v3 and 3.59 percent for ResNet-50, respectively. Because this class includes non-target plant species, it means that over 3% of our weed targets are incorrectly labeled as negative.

Species	Top-1 Accuracy (%)		Precision (%)		False-positive rate (%)	
	Inception-v3	ResNet-50	Inception-v3	ResNet-50	Inception-v3	ResNet-50
Chinee apple	85.3	**88.5**	**92.7**	91.0	**0.48**	0.61
Lantana	94.4	**95.0**	90.9	**91.7**	0.62	**0.55**
Parkinsonia	96.8	**97.2**	95.6	**97.9**	0.29	**0.13**
Parthenium	94.9	**95.8**	95.8	**96.7**	0.26	**0.21**
Prickly acacia	92.8	**95.5**	**93.4**	93.0	**0.43**	0.46
Rubber vine	**93.1**	92.5	**99.2**	99.1	**0.05**	0.05
Siam weed	**97.6**	96.5	94.4	**97.2**	0.38	**0.18**
Snakeweed	88.0	**88.5**	86.9	**90.9**	0.82	0.55
Negatives	97.2	**96.5**	96.5	**96.7**	3.77	3.59
Weighted average	95.1	**95.7**	95.1	**95.7**	2.16	**2.04**

Table 2. The average test classification accuracy, recall rate, precision, and false-positive rate across all five cross-validated folds for both Inception-v3 and ResNet-50. The statistic from the best-performing network is emboldened for each species. Equations for the computation of each metric are provided above.

VI. Comments

A. The paper lucidly explains the way data was collected and prepared, and the network architectures used. The results provide an insightful comparative analysis of Inception-v3 and Resnet - 50.

B. Weed control is a global topic that has gotten a lot of press in recent years. Several attempts have been made to identify weeds in crops. We concentrated on image classification because our scope was confined to the networks presented in Chapter 7. Semantic segmentation, on the other hand, might be a better method to address the challenge of weed detection in crops. Jie You, Wei Liu, and Joonwhoan Lee's research, 'A DNN-based semantic segmentation for detecting weed and crop,' employs semantic segmentation over the architectures used in this paper to better categorize weeds across crops.

C. This paper has inspired further research into the classification of weeds, concluding that ResNet-50 and Inception-v3 have resulted in better accuracy than the other convolutional neural network architectures for large datasets. However, the VGG16 model gives the best results for the DeepWeeds dataset as they had the least convolutional layers, which was adequate for small datasets. A drawback of the VGG16 model is that it uses more parameters than ResNet-50, which may lead to overfitting.

Assignment 1 Team Study of CNN								
Team #18								
Student name: Aniruddha Anand Damle	worked on literature	worked on implementation (data, platform, test run, debug, compatibility…)	generated results (run results, result data processing, presenting results	wrote report (Intro, method, result, discussions, …)	other significant contributions	peer approval 1	peer approval 2	peer approval 3
specific & detailed evidence is required to support claims of contributions (make reference to specific paragrphs, equation #, figure #, code line #'s sections, etc…)	1. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. 2. Analysis of ResNet and GoogleNet models for malware detection	N/A	N/A	Section 4, Section 5, Comments and IEEE formatting	Presented paper on malware detection to other team members to convince them to consider this paper for our CNN Team study assignment. In order to answer the problem in the chosen paper, I compared the outcomes of the semantic segmentation approach.	N/A	Approved	Approved
Student name: Prakriti Biswas	worked on literature	worked on implementation (data, platform, test run, debug, compatibility…)	generated results (run results, result data processing, presenting results	wrote report (Intro, method, result, discussions, …)	other significant contributions	peer approval 1	peer approval 2	peer approval 3
specific & detailed evidence is required to support claims of contributions (make reference to specific paragrphs, equation #, figure #, code line #'s sections, etc…)	1. Rethinking the Inception Architecture for Computer Vision 2. Visual Place Recognition by Spatial Matching of High-Level CNN	N/A	N/A	Abstract, Discussion Questions 1 and 2 (Section 1, 2, and 3)	Presented paper on Inception and VGG16 to other team members to convince them to review the particular research papers. Compared VGG16 with ResNet-50 models to gauge the importance of paper presented for review.	Approved	N/A	Approved
Student name: Aditya Kaduskar	worked on literature	worked on implementation (data, platform, test run, debug, compatibility…)	generated results (run results, result data processing, presenting results	wrote report (Intro, method, result, discussions, …)	other significant contributions	peer approval 1	peer approval 2	peer approval 3
specific & detailed evidence is required to support claims of contributions (make reference to specific paragrphs, equation #, figure #, code line #'s sections, etc…)	1. DeepWeeds: A Multiclass Weed Species Image Dataset for Deep Learning (Reference selected for study.)	N/A	N/A	Sections 3 and 6	Presented the reference I individually researched to other team members, in an attempt to convince them to use this paper for the study. Also, helped team members understand different network architectures during the initial phase of the study.	Approved	Approved	N/A