weedNet: Dense Semantic Weed Classification Using Multispectral Images and MAV for Smart Farming

Inkyu Sa, Zetao Chen, Marija Popović, Raghav Khanna, Frank Liebisch, Juan Nieto, and Roland Siegwart

Abstract—Selective weed treatment is a critical step in autonomous crop management as related to crop health and yield. However, a key challenge is reliable and accurate weed detection to minimize damage to surrounding plants. In this letter, we present an approach for dense semantic weed classification with multispectral images collected by a micro aerial vehicle (MAV). We use the recently developed encoder–decoder cascaded convolutional neural network, SegNet, that infers dense semantic classes while allowing any number of input image channels and class balancing with our sugar beet and weed datasets. To obtain training datasets, we established an experimental field with varying herbicide levels resulting in field plots containing only either crop or weed, enabling us to use the normalized difference vegetation index as a distinguishable feature for automatic ground truth generation. We train six models with different numbers of input channels and condition (fine tune) it to achieve ∼0.8 F1-score and 0.78 area under the curve classification metrics. For the model deployment, an embedded Graphics Processing Unit (GPU) system (Jetson TX2) is tested for MAV integration. Dataset used in this letter is released to support the community and future work.
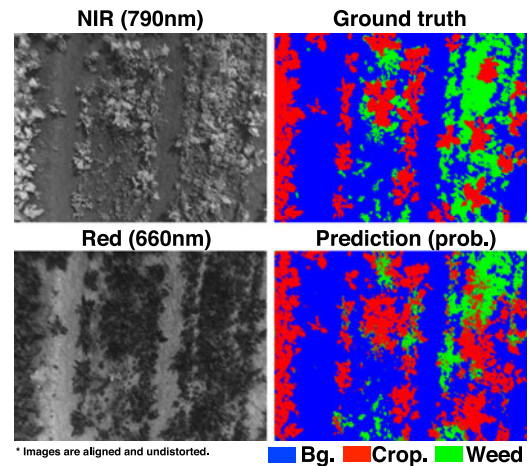
Fig. 1. NIR (top-left) and Red channel (bottom-left) input images with ground truth (top-right). Bottom-right is the probability output from our dense semantic segmentation framework.

I. INTRODUCTION

TO SUSTAIN a growing worldwide population with sufficient farm produce, new smart farming methods are required to increase or maintain crop yield while minimizing environmental impact. Precision agriculture techniques achieve this by spatially surveying key indicators of crop health and applying treatment, e.g., herbicides, pesticides, and fertilizers, only to relevant areas. Here, robotic systems can be often used as flexible, cost-efficient platforms replacing laborious manual procedures.

Specifically, weed treatment is a critical step in autonomous farming as it directly associates with crop health and yield [1]. Reliable, and precise weed detection is a key requirement for effective treatment as it enables subsequent processes, e.g., selective stamping, spot spraying, and mechanical tillage, while minimizing damage to surrounding vegetation. However, accurate weed detection presents several challenges. Traditional object-based classification approaches are likely to fail due to unclear crop-weed boundaries, as exemplified in Fig. 1. This aspect also impedes manual data labeling which is required for supervised learning algorithms.

To address these issues, we employ a state-of-the-art dense (pixel-wise) Convolutional Neural Network (CNN) for segmentation. We chose this CNN because it can predict relatively faster than other algorithms while maintaining competitive performance. The network utilizes a modified VGG16 [2] architecture (i.e., dropping last two fully-connected layers) as an encoder, and a decoder is formed with upsampling layers that are counterparts for each convolutional layer in the encoder.

Our CNN encodes visual data as low-dimensional features capturing semantic information, and decodes them back to higher dimensions by up-sampling. To deal with intensive manual labeling tasks, we maneuver a micro aerial vehicle (MAV) with a downward-facing multispectral camera over a designed

field with varying herbicide levels applied to different crop field areas. To this end, we obtain 3 types of multispectral image datasets which contain (i) only crop, (ii) only weed, and (iii) both crop and weed. For (i) and (ii), we can easily create ground truth by extracting Normalized Difference Vegetation Index (NDVI) indicating vegetation cover. Unfortunately, we could not avoid manual labeling for (iii), which took about 60 mins per image (with 30 testing images). Given these training and testing datasets, we train 6 different models with varying input channels and training conditions (i.e., fine-tune) to see the impact of these variances. The trained model is deployed on an embedded GPU computer that can be mounted on a small-scale MAV.

The contributions of this system paper are:
- Release of pixel-wise labelled (ground-truth) sugar beet/weed datasets collected from a controlled field experiment,[1]
- A study on crop/weed classification using dense semantic segmentation with varying multispectral input channels.

Since the dense semantic segmentation framework predicts the probability for each pixel, the outputs can be easily used by high-level path planning algorithms, e.g., monitoring- [3] or exploration-based [4], for informed data collection and complete autonomy on the farm [5].

The remainder of this letter is structured as follows. Section II presents the state-of-the-art on pixel-wise semantic segmentation, vegetation detection using a MAV, and CNN with multispectral images. Section III describes how the training/testing dataset is obtained and details of our model training procedure. We present our experimental results in Section IV, and conclude the letter in Section V.

II. RELATED WORK

This section briefly reviews the state-of-the-art in deep segmentation models, general methods of detecting vegetation, and segmentation based on multispectral images.

A. Pixel-Wise Segmentation Using Deep Neural Network

The aim of the image segmentation task is to infer a human-readable class label for each image pixel, which is an important and challenging task. The most successful approaches in recent years rely on CNN. Early CNN-based methods perform segmentation in a two-step pipeline, which first generates region proposals and then classifies each proposal to a pre-defined category [6], [7]. Recently, fully Convolutional Neural Networks (FCNNs) have become a popular choice in image segmentation, due to their rich feature representation and end-to-end training [8], [9]. However, these FCNN-based methods usually have a limitation of low-resolution prediction, due to the sequential max-pooling and down-sampling operation. SegNet [10], the module that our weedNET is based on, is a recently proposed pixel-wise segmentation module that carefully addresses this issue. It has an encoder and a corresponding decoder network. The encoder network learns to compress the image into a lower-resolution feature representation, while the decoder net-

work learns to up-sample the encoder feature maps to full input resolution for per-pixel segmentation. The encoder and decoder networks are trained simultaneously and end-to-end.

B. MAV-Based Crop-Weed Detection

For smart farming applications, it is becoming increasingly important to accurately estimate the type and distribution of the vegetation on the field. There are several existing approaches which exploit different types of features and machine learning algorithms to detect vegetation [11]–[13]. Torres-Sánchez *et al.* [14] and Khanna *et al.* [15] investigate the use of NDVI and Excess Green Index (EGI) to automatically detect vegetation from soil background. Comparatively, Guo *et al.* [16] exploit spectral features from RGB images and decision tress to separate vegetation. A deeper level of smart farming is an automatic interpretation of the detected vegetation into classes of crop and weed. Pérez-Ortiz et al. [17] utilize multispectral image pixel values as well as crop row geometric information to generate features for classifying image patches into valuable crop, weed, and soil. Similarly, Peña *et al.* [18] exploit spatial and spectral characteristics to first extract image patches, and then use the geometric information of the detected crop rows to distinguish crops and weeds. In [19], visual features as well as geometric information of the detected vegetation are employed to classify the detected vegetation into crops and weeds using Random Forest algorithms [20]. All the above-mentioned approaches either directly operate on raw pixels or rely on a fixed set of handcrafted features and learning algorithms [21].

However, in the presence of large data, recent developments of deep learning have shown end-to-end learning approaches [22] outperforms traditional hand-crafted feature learning [6], [23]. Inspired by this, Mortensen *et al.* [24] proposed a CNN-based semantic segmentation approach to classify different types of crops and estimate their individual amount of biomass. Compared to their approach which operates on RGB images, the approach in this letter extract information from multispectral images using a different state-of-the-art per-pixel segmentation model.

C. Applications Using Multispectral Images

Multi-spectral images provide the possibility to create vegetation specific indices based on radiance ratios which are more robust under varying lighting conditions and therefore are widely explored for autonomous agriculture robotics [25]–[28]. In [29], images captured from a six band multi-spectral camera were exploited to segment different parts of sweet pepper plants. Similarly, in [18], the authors managed to compute weed maps in maize fields from multispectral images. In [30], multispectral images were used to separate sugar beets from a thistle. In our study, we apply a CNN-based pixel-wise segmentation model directly on the multispectral inputs to accurately segment crops from weeds.

As shown above, there are intensive research interests in agricultural robotics domain employing state-of-the-art CNN, multispectral images, and MAVs for rapid field scouting. Particularly, precise and fast weed detection is the key front-end module that

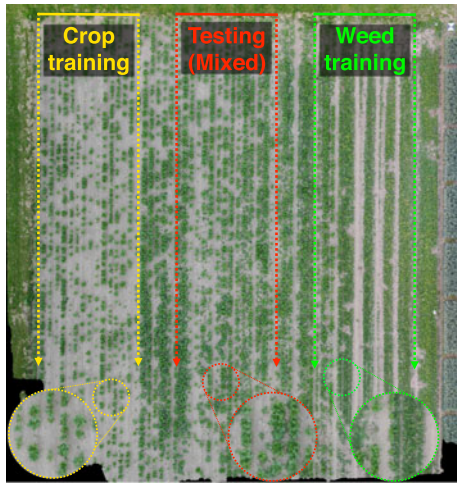[1]Available at: https://goo.gl/UK2pZq

Fig. 2. Aerial view of our controlled field with a varying herbicide levels. The maximum amount of herbicide is applied to the left crop training rows (yellow), and no herbicide is utilized for the right weed training rows (green). The middle shows mixed variants due to medium herbicide usage (red).

TABLE I
IMAGE DATASETS FOR TRAINING AND TESTING

| (NIR+Red+NDVI)	Crop	Weed	Crop-weed	Num. multispec
Training	132	243	—	375
Testing	—	—	90	90
Altitude (m)	2	2	2	—

Fig. 3. An input crop image (left), with NDVI is extracted using NIR+Red channels (mid.). An auto-threshold boundary detection method outputs clear edges between vegetation and others (shadows, soil, and gravel) (right).

can lead subsequent treatments to success. To our best knowledge, this work is the first trial applying a real-time ($\approx$2 Hz) CNN-based dense semantic segmentation using multispectral images taken from a MAV to agricultural robotics application while maintaining applicable performance. Additionally, we release our dataset utilized in this letter to support the community and future work since there are insufficient public available weed dataset [13].

III. METHODOLOGIES

In this section, we present our approaches to dataset acquisition and pre-processing through image alignment and NDVI extraction before outlining our model training method.

A. Dataset Acquisition

Dense manual annotation of crop/weed species is a challenging task, as shown in Fig. 1. Unlike urban street (e.g., [31] and [32]) or indoor scenes [33] that can easily be understood and inferred intuitively, plant boundaries in our datasets are difficult to distinguish and may require domain-specific knowledge or experience. They also require finer-selection tools (e.g., pixel-level selection and zoom-in/out) rather than polygon-based outline annotation with multiple control points. Therefore, it may be difficult to use coarse outsourced services [34].

To address this issue, we designed the 40 m $\times$ 40 m weed test field shown in Fig. 2. We applied different levels of herbicide to the field; max, mid, and min, corresponding to left (yellow), mid (red), and right (green) respectively. Therefore, as expected, images from the left-to-right field patches contain crop-only, crop/weed, and weed-only, respectively. We then applied basic automated image processing techniques to extract NDVI from the left and right patch images, as described in the following sub-section. The crop-weed images could only be annotated manually following advice from crop science experts. This process took on average about 60 mins/image. As shown in Table I, we annotated 132, 243, and 90 multispectral images of
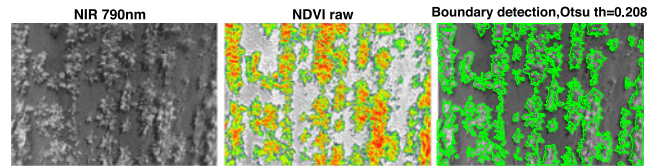
crops, weeds, and crop-weed mixtures. Each training image/test image consisted of near-infrared (NIR, 790 nm), Red channel (660 nm), and NDVI imagery.

B. Data Pre-Processing

For image acquisition, we use a Sequoia multispectral sensor that features four narrow band global shutter imagers (1.2 MP), and one rolling shutter RGB camera (16 MP). From corresponding NIR and Red images, we extract the NDVI, given by $NDVI = \frac{(NIR-Red)}{(NIR+Red)}$. This vegetation index clearly indicates the difference between soil and plant, as exemplified in Fig. 3 (NDVI raw).

1) Image Alignment: To calculate indices, we performed basic image processing for the NIR and Red images through image undistortion, estimation of geometric transformation $\in$ SE3 using image correlation, and cropping. Note that the processing time for these procedures is negligible since these transformations need to be computed only once for cameras attached rigidly with respect to each other. It is also worth mentioning that we could not align the other image channels, e.g., Green and Red Edge, in the same way due to the lack of similarities. Furthermore, it is difficult to match them correctly without estimating the depth of each pixel accurately. Therefore, our method assumes that the camera baseline is much smaller than the distance from the ground and camera ($\sim$two orders of magnitude). We only account for camera intrinsics, and do not apply radiometric and atmospheric corrections.

2) NDVI Extraction: We then applied a Gaussian blur to the aligned images (threshold=1.2), followed by a sharpening procedure to remove fine responses (e.g., shadows, small debris). An intensity histogram clustering algorithm, Otsu's method [35], is used for threshold selection on the resultant image and blob detection is finally executed with the minimum of 300 connected pixels. Fig. 3 shows the detected boundary of a crop image and each class is labeled as {Bg, Crop, Weed}={0,1,2}. Bg indicates background (mostly soil but not necessary), Crop and Weed are the sugar beet plant and weed class respectively.
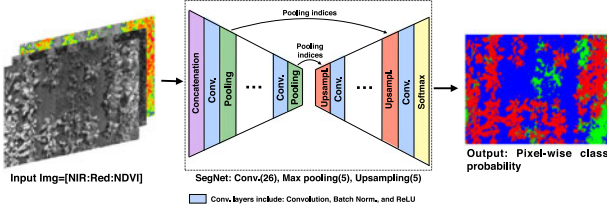
Fig. 4. An encoder-decoder cascaded dense semantic segmentation framework [10]. It has 26 convolution layers followed by ReLU activation and 5 max-pooling for the encoder (first half) and 5 up-sampling for the decoder (second half) linked with pooling indices. The first concatenation layer allows for any number of input channels. The output depicts the probability of 3 classes.

C. Dense Semantic Segmentation Framework

The annotated images are fed into SegNet, a state-of-the art dense segmentation framework shown in Fig. 4. We retain the original network architecture (i.e., VGG16 without fully-connected layers and additionally upsample layers for each counterpart for max-pooling) [10] and we only present our modifications.

Firstly, the frequency of appearance (FoA) for each class is adapted based on our training dataset for better class balancing [36]. This is used to weigh each class inside the neural network loss function and requires careful tuning. For example, as the weed class appears less frequently than bg and crop, its FoA is lower in comparison. If a false-positive or false-negative is detected in weed classification (i.e., a pixel is incorrectly classified as weed), then the classifier is penalized more than for the other classes. A class weight can be written as:

$$w_C = \frac{\widetilde{FoA(c)}}{FoA(c)} \quad (1)$$

$$FoA(c) = \frac{I_C^{\text{Total}}}{I_C^j} \quad (2)$$

where $\widetilde{FoA(c)}$ is the median of $FoA(c)$, I_C^{Total} is the total number of pixels in class c, and I_C^j is the number of pixels in the jth image where class c appears, with $j \in \{1, 2, 3...N\}$ as the image sequence number.

Secondly, we implemented a simple input/output layer that reads images and outputs them to the subsequent concatenation layer. This allows us to feed any number of input images to the network, which is useful for hyperspectral image processing [37].

IV. EXPERIMENTAL RESULTS

In this section, we present our experimental setup, followed by a qualitative and quantitative evaluation of our proposed approach. We also demonstrate a preliminary performance evaluation of our model deployed on an embedded computer.

A. Experimental Setup

As shown in Fig. 2, we cultivated a 40 m× 40 m test sugar beet field with varying herbicide levels applied for automated ground-truth acquisition following the procedures in Section III-B2.
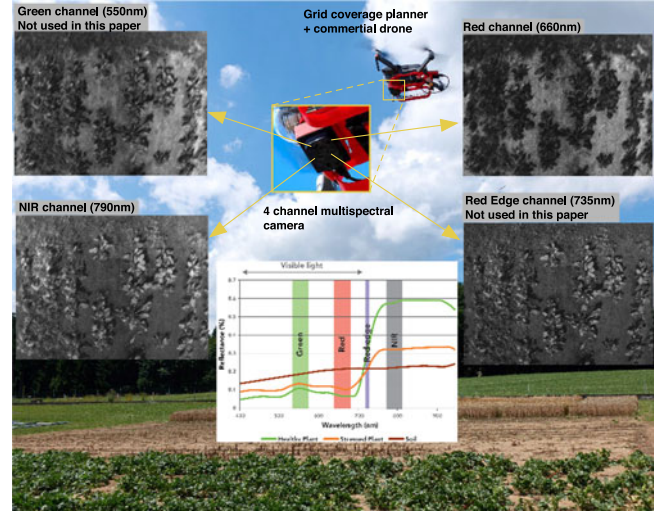

Fig. 5. Data collection setup. A DJI Mavic flies over our experimental field with a 4-band multispectral camera. In this letter, we consider only NIR and Red channels due to difficulties in image registration of other bands. The graph illustrates the reflectance of each band for healthy and sick plants, and soil (image courtesy of Micasense[2]).

A downward-facing Sequoia multispectral camera is mounted on a commercial MAV, DJI Mavic, recording datasets at 1 Hz (Fig. 5). The MAV is manually controlled in position-hold mode assisted by GPS and internal stereo-vision system at 2 m height. It can fly around 15 ∼17 mins with an additional payload of 274.5 g, including an extra battery pack, Sequoia camera, radiation sensor, a DC-converter, and 3D-printed mounting gear. Once a dataset is acquired, the information is transferred manually to a ground station.

For model training, we use NVIDIA's Titan X GPU module on a desktop computer and a Tegra TX2 embedded GPU module with an Orbitty carrier board for model inference.

We use MATLAB to convert the collected datasets to the SegNet data format and annotate the images. A modified version of Caffe [38] with cuDNN processes input data using CUDA, C++ and Python 2.7. For model training, we set the following parameters: learning rate = 0.001, maximum iterations = 40,000 (640 epochs), batch size = 6, weight decay rate = 0.005 and Stochastic Gradient Descent (SGD) solver [39] is used for the optimization. The average model training time given the maximum number of iterations is 12 hrs. Fig. 6 shows the loss and average class accuracy over 40,000 iterations. This figure suggests that 10,000–20,000 maximum iterations are sufficient since there is a very subtle performance improvement beyond this.

B. Quantitative Results

For quantitative evaluation, we use a harmonic F1 score measure capturing both precision and recall performance as:

$$F1(c) = 2 \cdot \frac{\text{precision}_c \cdot \text{recall}_c}{\text{precision}_c + \text{recall}_c}$$

$$\text{precision}_c = \frac{TP_c}{TP_c + FP_c}$$

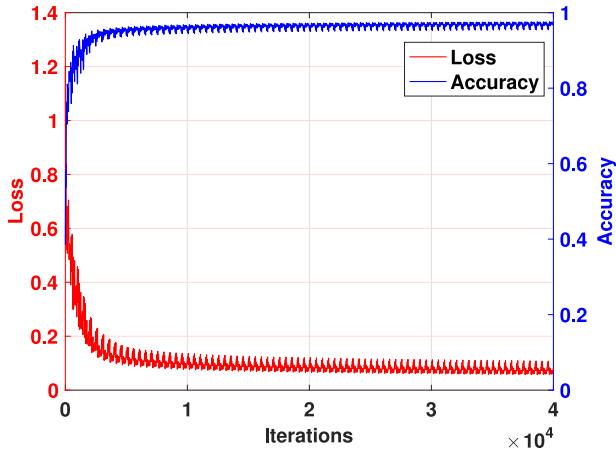$$\text{recall}_c = \frac{TP_c}{TP_c + FN_c} \quad (3)$$

Fig. 6. Loss and average class accuracy of three input channels with fine-tuning over various iterations. The maximum number of iterations is set to 40,000, which takes 12 hrs.
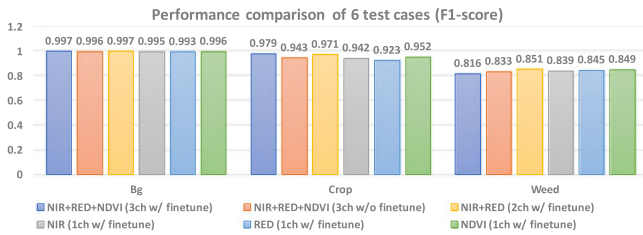

Fig. 7. F1-scores of 6 models per classes (horizontal axis). Higher values indicate the better detection performance. Note that the right bars show weed classification F1-score.
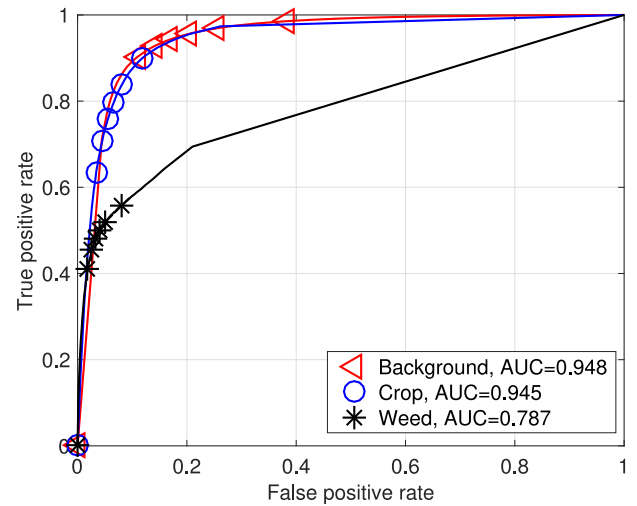

Fig. 8. AUC for our 3 channel model corresponding to dark blue bars (left-most) in Fig. 7. Note that the numbers do not match since these measures capture slightly different properties.

where precision$_c$ and TP_c indicate the precision and number of true positives, respectively, for class c. The same subscript convention is applied to other notation. Note that the output of SegNet are the probabilities of each pixel belonging to each defined class. In order to compute four fundamental numbers (i.e., TP, TN, FP, and FN), the probability is converted into a binary value. We simply assign the class label of maximum probability and compute precision, recall, accuracy, and F1-score. All models presented in this section are trained and tested with the datasets in Table I.

Fig. 7 shows the F1-scores of 6 different models trained with varying input data and conditions. There are three classes; Bg, Crop, and Weed. All models perform reasonably well (above 80% for all classes) considering the difficulty of the dataset. In our experiments, we vary two conditions; using a pre-trained model (i.e., VGG16) for network initialization (fine-tuning) and varying the number of input channels

As shown in Fig. 7, the dark blue and orange (with and without fine-tuning given 3 channel input images, respectively) showed that fine-tuning does not impact the output significantly in comparison with more general object detection tasks (e.g., urban scenes or daily life objects). This is mainly because our training datasets (sugar beet and weed images) have different appearance and spectra (660–790 nm) for the pre-trained model based on RGB ImageNet dataset [40]. There may be very few scenes that are similar to our training data. Moreover, the size of

dataset we have is relatively small compared with the pre-trained ImageNet (1.2 Million images).

The yellow, gray, light blue, and green bars in Fig. 7 present performance measures with varying numbers of input channels. We expected more input data to yield better results since the network captures more useful features that help to distinguish between classes. This can be seen by comparing the results of the 2 channel model (NIR+Red, yellow bar from Fig. 7) and the 1 channel model of NIR and Red (gray and light blue respectively). 2 channel model outperforms for both crop and weed classification. However, interestingly, the number of input data does not always guarantee performance improvement. For instance, the NIR+Red model surpasses the 3 channel model for weed classification performance. We are still investigating this and suspect that it could be due to i) NDVI image is produced based on NIR and Red images meaning that NDVI depends on those two images rather than capturing new information, ii) inaccurate image alignment of NIR and Red image channel on the edge of image where larger distortions exist than in the optical center. Inference performance with varying input data, is discussed in Section IV-E.

We also use area-under-the-curve (AUC) measures for quantitative evaluation. For example, Fig. 8 shows the AUC of 3 channel model. It can be seen that there is small performance variation. As these measures capture different classifier properties, they cannot be directly compared.

C. Qualitative Results

We perform a qualitative evaluation with our best performance model. For visual inspection, we present 7 instances of all input data, ground-truth, and network probability output in Fig. 9. Each row displays an image frame, with the first three columns showing the input of our 3 channel model. NDVI is displayed as heat map scale (colors closer to red depict higher response). The fourth column is annotated ground-truth, and the fifth is our probability output. Note that each class probability is
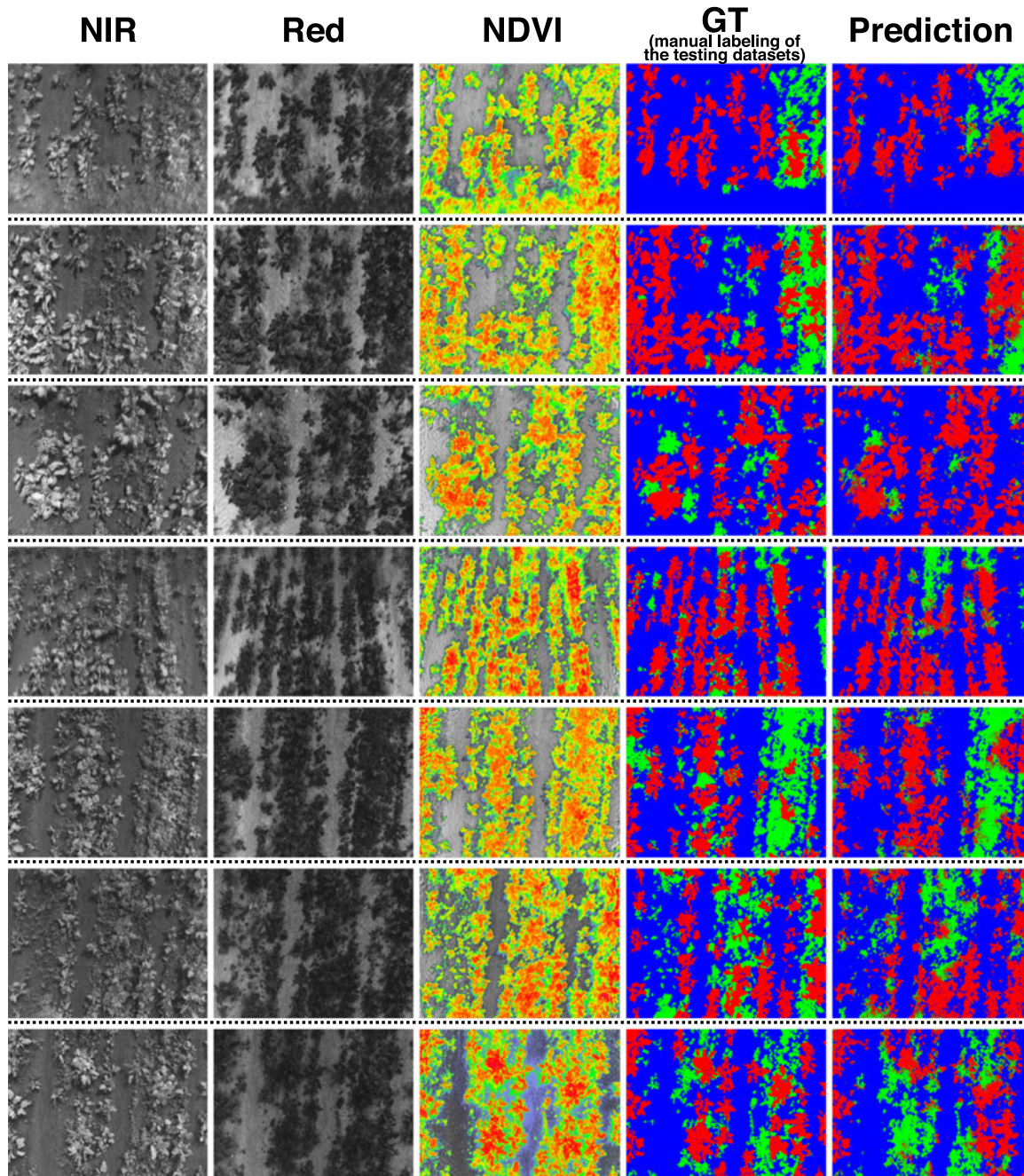
Fig. 9. The qualitative results of 7 frames (row-wise). The first three columns are input data to the CNN, with the fourth and fifth showing ground-truths and probability predictions. Because we map the probability of each class to the intensity of R, G, B for visualization, some boundary areas have mixed color.

color-coded as intensities of the corresponding color such that background, crop, and weed represent blue, red, and green, respectively. It can be seen that some boundary areas have mixed color (e.g., a pixel value of [0.4, 0.6, 0.7] that indicates probability of 0.4 background, 0.6 crop and 0.7 green). There are noticeable misclassification areas of crop detection in the last row and evident weed misclassification in the second and fifth. This mostly occurs when crops or weeds are surrounded by each other, implying that the network captures not only low-level features, such as edges or intensities, but also object shapes and textures.

D. Discussion, Limitations, and Outlook

We demonstrated a crop/weed detection pipeline that uses training/testing datasets obtained from consistent environmental conditions. Although it shows applicable qualitative and quantitative results, it is important to validate how well it handles scale variance, and its spatio-temporal consistency.

We study these aspects using an independent image in Fig. 10. The image was captured from a different sugar beet field in a month prior to the dataset we used (same altitude and sensor). It clearly shows that most of the crops are classified as weed (green) meaning our classifier requires more temporal training
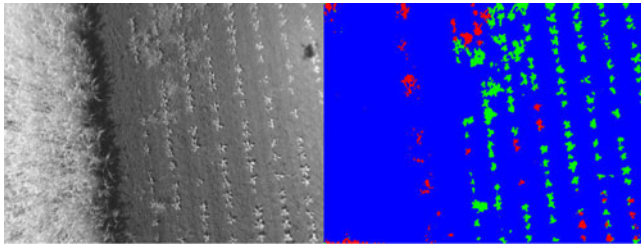
Fig. 10. Test image from a different growth stage of sugarbeet plants (left) and its output (right). The same color code is applied for each class as in Fig 9, i.e., blue=bg, red=crop, and green=weed.). This shows that our model reports a high false-positive rate because our training dataset has bigger crops and different type of weeds. Most likely, this is caused by a limited crop/weed temporal training dataset.
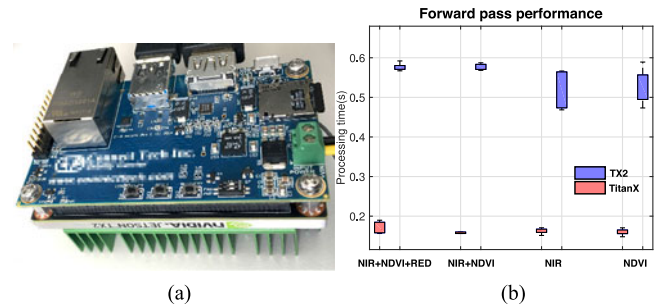

Fig. 11. (a) An embedded GPU module (190 g and 7.5 W) on which we deploy our trained model. (b) Network forward-pass processing time (y-axis) for different models (x-axis) using Tixan X (red) and Jetson TX2 (blue).

dataset. As shown in Fig. 9, we trained a model with larger crops and weeds images than in the new image. Additionally, there may be a different type of weed in the field that the model has not been trained for.

This exemplifies an open issue concerning supervised learning approaches. To address this, we require more training data covering multi-scale, wide-weed varieties over longer time periods to develop a weed detector with spatio-temporal consistency or smarter data augmentation strategies. Even though manually annotating each image would be a labor intensive task, we are planning to incrementally construct a large dataset in the near future.

E. Inference on an Embedded Platform

Recent developments in the deep CNN have played a significant role in computer vision and machine learning (especially object detection and semantic feature learning). However, there is still an issue of running trained models with relatively fast speed (2∼5 Hz) on a physically constrained device, which can be deployed on a mobile robot. To address this, researchers often utilize a ground station that has a decent GPU computing power with WiFi connection [41]. However, this may cause a large time delay, and it may be difficult to ensure wireless communication between a robot and ground station if coverage is somewhat limited.

Using an onboard GPU computer can resolve this issue. Recently an embedded GPU module, Jetson TX2, has been released that performs reasonably well as shown in Fig. 11(a). It has 2 GHz hexa-CPU cores, 1.3 GHz 256 GPU cores, and consumes 7.5 W while idle and 14 W for maximum utilization. Fig. 11(b) shows processing time comparison between Titan X (red) and TX2 (blue). We process 300 images using 4 models denoted in the x-axis. Titan X performs 3.6 times faster than TX2 but considering its power consumption ratio, 17.8 (250 W for Titan X maximum utilization), TX2 performs significantly well. Another interesting observation in Fig. 11(b) is that network forward-pass processing time does not affect the number of input channels much. This is because the first multi-convolution layer of all these models has the same filter size of 64. While varying number of inputs affects the contents of these low-level

features (e.g., captures different properties of a plant), the size remains identical.

Note that offline inference (i.e., loading images from stored files) is performed on TX2 since the multispectral camera only allows for saving images to a storage device. We are planning to integrate another hyperspectral camera (e.g., Xemia snapshot camera) for online and real-time object detection that can be interfaced with a control [42] or informative path planning [3] modules.

V. CONCLUSIONS

We demonstrated CNN-based dense semantic classification for weed detection with aerial multispectral images taken from an MAV. The encoder-decoder cascaded deep neural network is trained on a dataset obtained from a herbicide-controlled sugar beet field to address labor intensive labeling tasks. The data obtained from this field is categorized into images containing only crops or weeds, or a crop-weed mixture. For the homogeneous imagery data, vegetation is automatically distinguished by extracting NDVI from multispectral images and applying classic image processing for model training. For the mixed imagery data, we performed manual annotation taking ∼30 hours.

We trained 6 different models on varying numbers of input channels, training conditions, and evaluated them quantitatively using F1-scores and AUC as metrics. A qualitative assessment was then performed by a visual comparison of ground-truth with probability prediction outputs. Given the test dataset (mixed), the proposed approach reports an acceptable performance of ∼0.8 F1-score for weed detection. However, we found spatio-temporal inconsistencies in our model due to limitations in the dataset it was trained on.

We then deploy the model on an embedded system that can be carried by a small MAV, and compare its performance to a high-performance desktop GPU in terms of inference speed and accuracy. Our experimental results estimate that the proposed deep neural network system can run the high-level perception task at 1.8 Hz on the embedded platform which can be deployed on a MAV.

Finally, multispectral weed and crop images with the corresponding ground truth used in this letter are released for the

robotics community and to support future work in agricultural robotics domain.

Acknowledgment

The author would like to acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research and H. Zellweger for the management of the experimental field.

References

[1] D. Slaughter, D. Giles, and D. Downey, "Autonomous robotic weed control systems: A review," *Comput. Electron. Agric.*, vol. 61, no. 1, pp. 63–78, 2008.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv: 1409.1556, 2014.

[3] M. Popovic, T. Vidal-Calleja , G. Hitz, I. Sa, R. Y. Siegwart, and J. Nieto, "Multiresolution mapping and informative path planning for UAV-based terrain monitoring," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vancouver, BC, Canada, 2017.

[4] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon "next-best-view" planner for 3D exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1462–1468.

[5] I. Sa *et al.*, "Build your own visual-inertial odometry aided cost-effective and open-source autonomous drone," arXiv:1708.06652, 2017.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.

[7] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. Eur. Conf. Computer Vision*, Springer, 2014, pp. 297–312.

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFS," arXiv:1606.00915, 2016.

[9] J. Dai, K. He, and J. Sun, "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1635–1643.

[10] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[11] F. Liebisch *et al.*, "Automatic UAV-based field inspection campaigns for weeding in row crops," in *Proc. EARSeL SIG Imag. Spectrosc. Workshop*, Zurich, Switzerland, 2017.

[12] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "DeepFruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, no. 8, 2016, Art. no. 1222.

[13] S. Haug and J. Ostermann, "A crop/weed field image dataset for the evaluation of computer vision based precision agriculture tasks," in *Computer Vision—ECCV 2014 Workshops*. Zürich, Switzerland: Springer, 2014, pp. 105–116.

[14] J. Torres-Sánchez, F. López-Granados, and J. M. Peña, "An automatic object-based method for optimal thresholding in UAV images: Application for vegetation detection in herbaceous crops," *Comput. Electron. Agric.*, vol. 114, pp. 43–52, 2015.

[15] R. Khanna, M. Möller, J. Pfeifer, F. Liebisch, A. Walter, and R. Siegwart, "Beyond point clouds-3D mapping and field parameter measurements using UAVs," in *Proc. IEEE 20th Conf. Emerg. Technol. Factory Autom.*, 2015, pp. 1–4.

[16] W. Guo, U. K. Rage, and S. Ninomiya, "Illumination invariant segmentation of vegetation for time series wheat images based on decision tree model," *Comput. Electron. Agric.*, vol. 96, pp. 58–66, 2013.

[17] M. Pérez-Ortiz, J. Pena, P. A. Gutiérrez, J. Torres-Sánchez, C. Hervás-Martínez, and 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.*, vol. 37, pp. 533–544, 2015.

[18] J. M. Peña, J. Torres-Sánchez, A. I. de Castro, M. Kelly, and F. López-Granados, "Weed mapping in early-season maize fields using object-based analysis of unmanned aerial vehicle (UAV) images," *PloS One*, vol. 8, no. 10, 2013, Art. no. e77151.

[19] P. Lottes, R. Khanna, J. Pfeifer, R. Siegwart, and C. Stachniss, "UAV-based crop and weed classification for smart farming," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2017, pp. 3024–3031.

[20] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[21] I. Sa *et al.*, "Peduncle detection of sweet pepper for autonomous crop harvesting—Combined Color and 3-D Information," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 765–772, Apr. 2017.

[22] J. Hwangbo, I. Sa, R. Siegwart, and M. Hutter, "Control of a quadrotor with reinforcement learning," *IEEE Robot. Autom. Lett.*, vol. 2, no. 4, pp. 2096–2103, Oct. 2017.

[23] M. Di Cicco, C. Potena, G. Grisetti, and A. Pretto, "Automatic model based dataset generation for fast and accurate crop and weeds detection," arXiv:1612.03019, 2016.

[24] A. K. Mortensen, M. Dyrmann, H. Karstoft, R. N. Jørgensen, and R. Gislum, "Semantic segmentation of mixed crops using deep convolutional neural network," in *Proc. Int. Conf. Agric. Eng.*, 2016.

[25] C. McCool, I. Sa, F. Dayoub, C. Lehnert, T. Perez, and B. Upcroft, "Visual detection of occluded crop: For automated harvesting," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 2506–2512.

[26] A. Lucieer, Z. Malenovskỳ, T. Veness, and L. Wallace, "Hyperuas—Imaging spectroscopy from a multirotor unmanned aircraft system," *J. Field Robot.*, vol. 31, no. 4, pp. 571–590, 2014.

[27] R. Khanna, I. Sa, J. Nieto, and R. Siegwart, "On field radiometric calibration for multispectral cameras," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 6503–6509.

[28] C. Hung, J. Nieto, Z. Taylor, J. Underwood, and S. Sukkarieh, "Orchard fruit segmentation using multi-spectral feature learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 5314–5320.

[29] C. Bac, J. Hemming, and E. Van Henten, "Robust pixel-based classification of obstacles for robotic harvesting of sweet-pepper," *Comput. Electron. Agric.*, vol. 96, pp. 148–162, 2013.

[30] F. J. Garcia-Ruiz, D. Wulfsohn, and J. Rasmussen, "Sugar beet (Beta vulgaris L.) and thistle (Cirsium arvensis L.) discrimination based on field spectral data," *Biosyst. Eng.*, vol. 139, pp. 1–15, 2015.

[31] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *Int. J. Robot. Res.*, vol. 32, pp. 1231–1237, 2013.

[32] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

[33] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suit," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 567–576.

[34] A. Sorokin and D. Forsyth, "Utility data annotation with Amazon mechanical turk," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2008, pp. 1–8.

[35] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[36] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2650–2658.

[37] R. Khanna, I. Sa, J. Nieto, and R. Siegwart, "On field radiometric calibration for multispectral cameras," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2017, pp. 6503–6509.

[38] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," arXiv:1408.5093, 2014.

[39] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT'2010*, Springer, 2010, pp. 177–186.

[40] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[41] A. Giusti *et al.*, "A machine learning approach to visual perception of forest trails for mobile robots," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 661–667, Jul. 2016.

[42] M. Kamel, T. Stastny, K. Alexis, and R. Siegwart, "Model predictive control for trajectory tracking of unmanned aerial vehicles using robot operating system," in *Robot Operating System (ROS): The Complete Reference*, vol. 2, A. Koubaa, Ed. New York, NY, USA: Springer, 2017.