Article

A System for Weeds and Crops Identification – Reaching over 10 FPS on Raspberry Pi with the Usage of MobileNets, DenseNet and Custom Modifications.

Łukasz Chechliński [1], Barbara Siemiątkowska [1] and Michał Majewski [2]

[1] Warsaw University of Technology, Warsaw, Poland
[2] MCMS Warka Ltd., Warka, Poland
* Correspondence: l.chechlinski@mchtr.pw.edu.pl

Abstract: Automated weeding is an important research area in agrorobotics. Weeds can be removed mechanically or with the precise usage of herbicides. Deep Learning techniques achieved state of the art results in many computer vision tasks, however their deployment on low-cost mobile computers is still challenging. These paper present an advanced version of the system presented in [1]. The described system contains several novelties, compared both with its previous version and related work. It is a part of a project of the automatic weeding machine, developed by Warsaw University of Technology and MCMS Warka Ltd. The obtained model reaches satisfying accuracy at over 10 FPS on the Raspberry Pi 3B+ computer. It was tested for four different plant species at different growth stadiums and lighting conditions. The system performing semantic segmentation is based on Convolutional Neural Networks. Its custom architecture mixes U-Net, MobileNets, DenseNet and ResNet concepts. Amount of needed manual ground truth labels was significantly decreased by the usage of knowledge distillation process, learning final model to mimic an ensemble of complex models on the large database of unlabeled data. Further decrease of the inference time was obtained by two custom modifications: in the usage of separable convolutions in DenseNet block and in the number of channels in each layer. In the authors' opinion, described novelties can be easily transferred to other agrorobotics tasks.

Keywords: Automated Weeding; Mobile Convolutional Neural Netowrks, Semantic Segmentation

1. Introduction

Weeds are considered to be one of the biggest problems in agronomy. Their adverse effect is widely known - they reduce crop yields, serve as hosts for crop diseases and also produce toxic substances [2]. Manual removal of weeds is labour-intensive, while the use of chemicals has long-term environmental consequences [3,4]. It seems that automatic mechanical weeding (or eventually precise usage of herbicides) is the solution to the problem. In recent years, there has been a significant development in the field of agricultural robotics. Automation and robotisation of agriculture allow us to reduce the number of people needed to carry out work in this sector, especially for seasonal work. Appropriate devices, differing in their degree of sophistication, cost, effectiveness and work efficiency, are described in many scientific publications.

The Greenbot [5] is autonomous robot developed for professionals in India. It can uproot weeds in a grape field, using a high speed blade. The robot is controlled by a Raspberry Pi 2 with the usage of camera, GPS and obstacle detecting sensor. The power source for this robot is provided through a solar panel along with a rechargeable battery, making it an environment-friendly device. Vitirover [6] is an autonomous robot designed to cut the grass and weeds between grape vines, it uses a solar panel to produce energy. A vision system is used for detecting the presence of weeds. The GPS antenna provides the coordinates for navigation of the robot. The robot OZ, described in agriculture magazines, is an autonomous robot, providing for fully mechanical weed control. The robot is equipped with lasers and cameras. It moves autonomously and detects different types of plants. Automated weeding robots are often very costly. From economic reasons, application of robotics system is justified only in countries with high labour costs. The system described in this paper is a part of a solution dedicated to emerging markets, prepared by MCMS Warka Ltd. in cooperation with the Faculty of Mechatronics of Warsaw University of Technology.

All agriculture robots need effective vision system. Plant identification has to be both accurate and rapid. The paper [7] presents the first systematic review of computer vision techniques for plant identification. The traditional vision system consists of two parts: features detection and classification. In [8,9] the scale-invariant feature transform (SIFT) is proposed, the histogram of oriented gradient (HOG) is described in [10]. A shape [11] is vital for object identification. Shape descriptors are usually combined with features like texture [12]. Many of commercially available automatic weeding solutions are based on color segmentation [13,14]. Such an approach does not enable distinguishing weeds from crops in a case when they have an equal size or they are adjoined.

The usage of convolutional networks (CNN) is a trend in the development of the next generation of weeding machines [15]. In [16] a hybrid model of AlexNet and VGGNET is proposed for crop-weed classification. It is inspired by the organization of animal visual cortex and is used for processing images. CNN allows learning spatial hierarchies of features, from low- to high-level patterns and typically consists of three types of layers: convolution, pooling, and fully connected layers. The convolution layer is an essential part of CNN. The architecture of this layer depends on the number of filters (convolution), the kernel size and the stride. Pooling layers combine the outputs of neuron clusters into a single neuron and reduce the dimensions of the data. Convolution and pooling layers, perform feature extraction. A fully connected layer classifies the image based on the features. It is known that learning the features through CNN can provide better results than conventional solutions. However, these solutions rely on a massive amount of training data, and the training process is very long. In order to improve image recognition process, new architectures of CNN have been introduced. In MobileNets [17] the operation of the convolution is divided into spatial and inter-channel parts, and the number of required calculations have been reduced several times. ResNet [18] solves the vanishing gradient issue. The core idea of ResNet is introducing a so-called "identity shortcut connection" that skips one or more layers. Dense Convolutional Network (DenseNet) [19] connects each layer to every other layer in a block in a feed-forward fashion. U-Net [20] was first developed for a biomedical image segmentation tasks, but later the architecture was succesfully applied to other domains. PSPNet (Pyramid Scene Parsing Network) [21] is the champion of ImageNet Scene Parsing Challenge 2016. By using Pyramid Pooling Module, with different-region-based context aggregated, PSPNet provides a superior framework for pixel-level prediction tasks.

Automated weeding is a popular agrorobotics research topic, but to our best knowledge this is the first work describing usage of modern mobile CNNs and a large database of unlabelled data in this domain.

2. Overview of the weeding machine

The weeding machine will be mounted to a tractor, which will result in a price lower than in case of standalone solutions. Computational power will be provided by a cheap Raspberry Pi 3B+ microcomputer. The machine must be user-friendly and suitable for low crops growing in rows. Several modules can be connected together to enable weeding of multiple rows. Depending on the selected tool weeds can be removed mechanically, with the precise usage of herbicides or thermally (with the usage of hot water vapor). Crops growing in rows cannot be damaged, at least 90% of weeds must be removed. The segmentation must be performed with a rate of at least 10 FPS to enable smooth device control. With the considered working velocity of 4 km/h, the images will strongly overlap.

The device will be mounted behind the tractor, a weeding tool will be mounted after the camera. The machine must be dust-proof, especially in case of higher work velocities. Weeding tool is placed in a known distance from the camera field of view, so constant measurement of machine movement is required. This measurement will be done by the fusion of signals from the measuring wheel and visual odometry (from multiple modules in case of multirow weeding). The odometry error must not be larger than 1cm at the distance of 1m, which is an approximate distance between the camera and the weeding tool.

Camera calibration, both internal and with the weeding tools, will be done at the machine production stage. However, camera to ground distance is depended on the machine-to-tractor mounting point and ground unevenness. Usage of ultrasound sonars will enable to measure this parameter online.

The system of crops and weeds semantic segmentation assigns a class probability for each pixel. Control of the weeding tool is based on the segmentation output, odometry system and calibration of all subsystems. The following part of the paper will describe only the semantic segmentation, the rest of the components are beyond its scope.

3. The methodology of plant segmentation system development

The segmentation method has to assign probabilities of three classes (crops, weeds, and ground) to each pixel. Input resolution equals 640x480 pixels, and the input camera image covers a width of a single crop row. Acquisition of exactly such images was performed in the season 2018, 17 training and 7 test videos were obtained. Larger dataset (containing 92 training and 34 test videos) was acquired in the season 2017, but with a different hardware configuration (another camera model, multiple crop rows in the field of view, another input resolution), so videos were manually cropped and scaled.

Each video contains images from the ride along a single crop row. Four crop species (beet, cauliflower, cabbage, strawberry) at different growth stages are present. The camera is facing down the ground. Sample images from both 2017 and 2018 seasons are presented in figure 1.

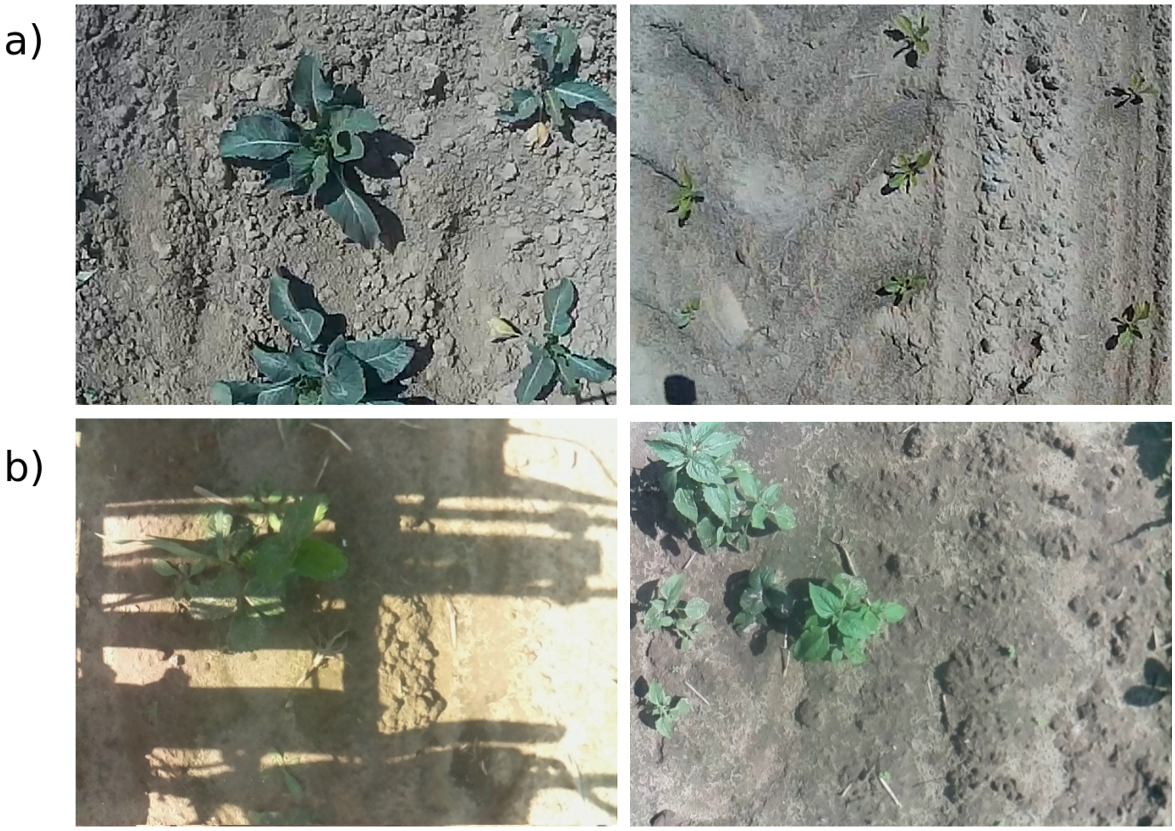

Figure 1. Example images from (**a**) the season 2017 (manually cropped and scaled) and (**b**) 2018.

Manual ground truth labels, assigning each pixel to one of three classes, were prepared for selected training images from both seasons, with the usage of dedicated aiding software. 261 labels were prepared for the season 2017 and 600 for the season 2018. A sample image with its ground

Peer-reviewed version available at *Sensors* **2019**, *19*, 3787; doi:10.3390/s19173787

truth label is presented in figure 2. The preparation of manual labels is time-consuming, exact borders between classes are ambiguous. Moreover, the arduous character of this work causes experts' inconsequences such as incorrect labels in less important places (plants at image borders, the ground between leaves of the same plant).

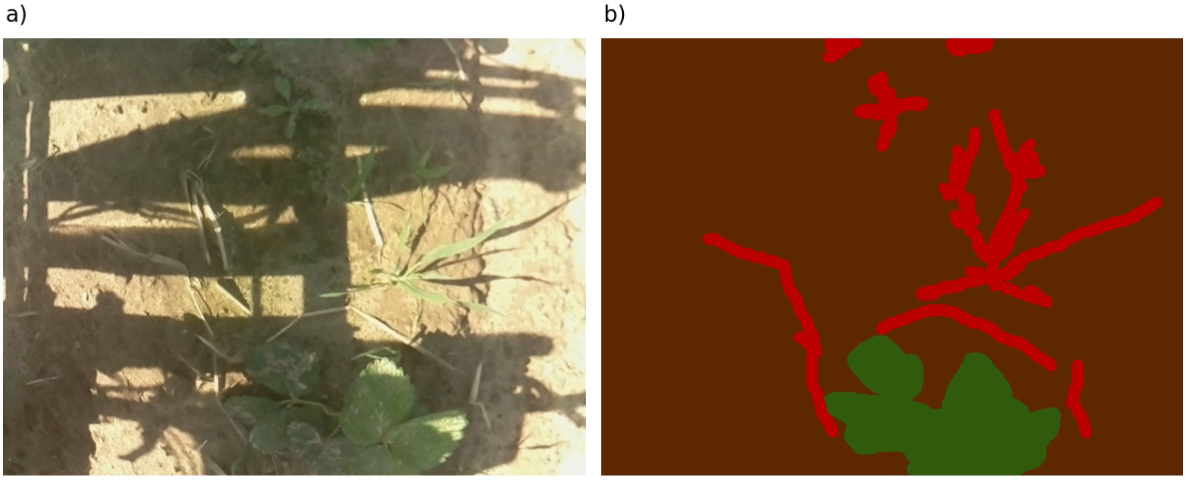

Figure 2. An example image (**a**) and the corresponding manual ground truth label (**b**). Crops are marked green, weeds are marked red and the ground is marked brown.

Training of the U-net based, semantic segmentation model was conducted with the usage of manual ground truth labels. This model is presented in figure 3 and is called the stationary, in opposition to the final mobile model. The number of channels depends on the parameter α (like in MobileNets the number of channels in selected layers is multiplied by α, details are presented in figure 3). The input resolution is reduced to 320x240 pixels (which equals about 2.3 mm/pixel on the ground level) and the output resolution was reduced to 80x60 pixels (about 9.2 mm/pixel). Such accuracy is satisfying in the automatic weeding task. Pixels on a border between classes were not included during the stationary model training. Data augmentation techniques were used, including changing images brightness and saturation, as well as horizontal and vertical image flipping.

Predictions of a single model are generally correct, but they contain some noise. It was reduced with the use of ensemble learning, calculating final probability distribution based on predictions of several independent model instances. A simple version of ensemble learning was chosen – first 50 stationary model instances were trained, and their results were averaged. A number of instances was chosen experimentally, in such a way that an output of the first half of the ensemble (25 instances) does not differ significantly from the output of the second half. Sample predictions of a single model and the models' ensemble for different input images are presented in figure 4.

The ensemble of stationary models will be a teacher in the knowledge distillation process. Its output class probability predictions for each pixel will be a label for a student mobile neural network. It means that the student mimics the entire output distribution of the teacher, not only the index of the most probable class. Such labels were calculated for all video frames from the season 2018 (about 16 000 training and 8 000 test images) and for uniformly sampled frames for the season 2017 (about 10 000 training and 10 000 test images).

The mobile model combines U-net, MobileNetsv2 and DenseNet architectures, its scheme is presented in figure 5. Transposed convolutions were replaced with activation map scaling, due to the lack of the separable transposed convolution implementation in optimized software frameworks. A block of convolutional layers is a basic element of the DenseNet architecture. Its adaptation for mobile devices will be called DenseNet-Mobile. A trivial version of this adaptation is presented in figure 6. Simple replacement of a full convolution with a separable one results in multiple calculations of several depthwise convolutions. To avoid this, a custom solution was designed, in which the depthwise convolution is calculated once for each channel. Simultaneously, earlier reduction of resolution is

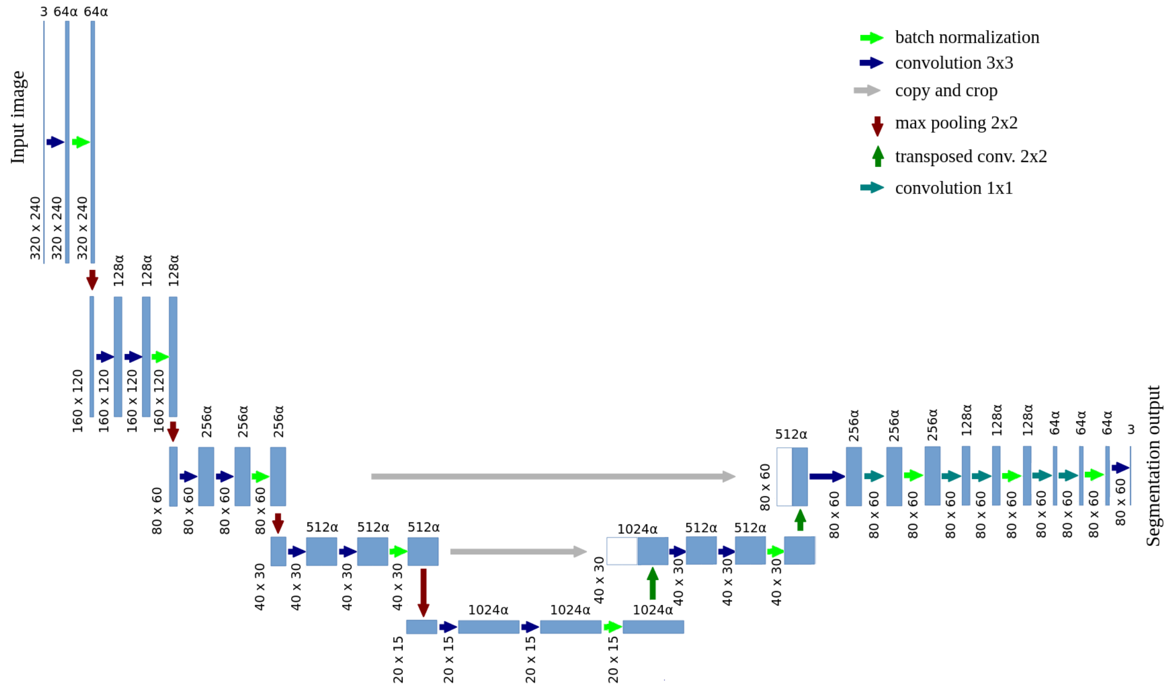

Figure 3. A scheme of the stationary, U-net based model, performing plants semantic segmentation.

possible in the case of resolution-reducing blocks, which additionally reduces the inference time. A scheme of such block is presented in figure 7. During inference of a mobile model on Raspberry Pi input was cropped to 320x112 pixels, which results in output resolution of 80x28 pixels. Such cropping allows to make prediction for each part of the crop row twice even at the machine velocity of about 4.6 km/h.

In the developed mobile architecture the number of channels in each layer depends on parameters A, B and C (according to figure 5). The MobileNets architecture originally introduces a parameter α, allowing to balance between model accuracy and inference time by multiplying the number of channels in each layer by this factor. In a variant directly based on MobileNets parameters A, B and C are defined as follows:

$$\begin{aligned} f_{MobileNet}(x) &= min(1, \lfloor x \cdot \alpha \rfloor) \\ A &= f_{MobileNet}(16) \\ B &= f_{MobileNet}(32) \\ C &= f_{MobileNet}(64) \end{aligned} \tag{1}$$

However, tests on the Raspberry Pi 3B+ microcomputer showed that such a definition of the number of channels results in nonmonotonic inference time in function of the α parameter and in nonfunctional dependency between model accuracy and inference time. This is caused by the high efficiency of inference for layers which number of channels is a power of two or a sum of few powers of two. For example, reducing the number of channels from $A = 16$, $B = 32$, $C = 64$ to $A = 15$, $B = 31$, $C = 62$ results in increase of the inference time. There are few α values that allow keeping an effective number of channels in each layer. To overcome this problem, a custom formula was introduced, which defines the number of channels in each layer as a multiple of 8. The value of 8 was

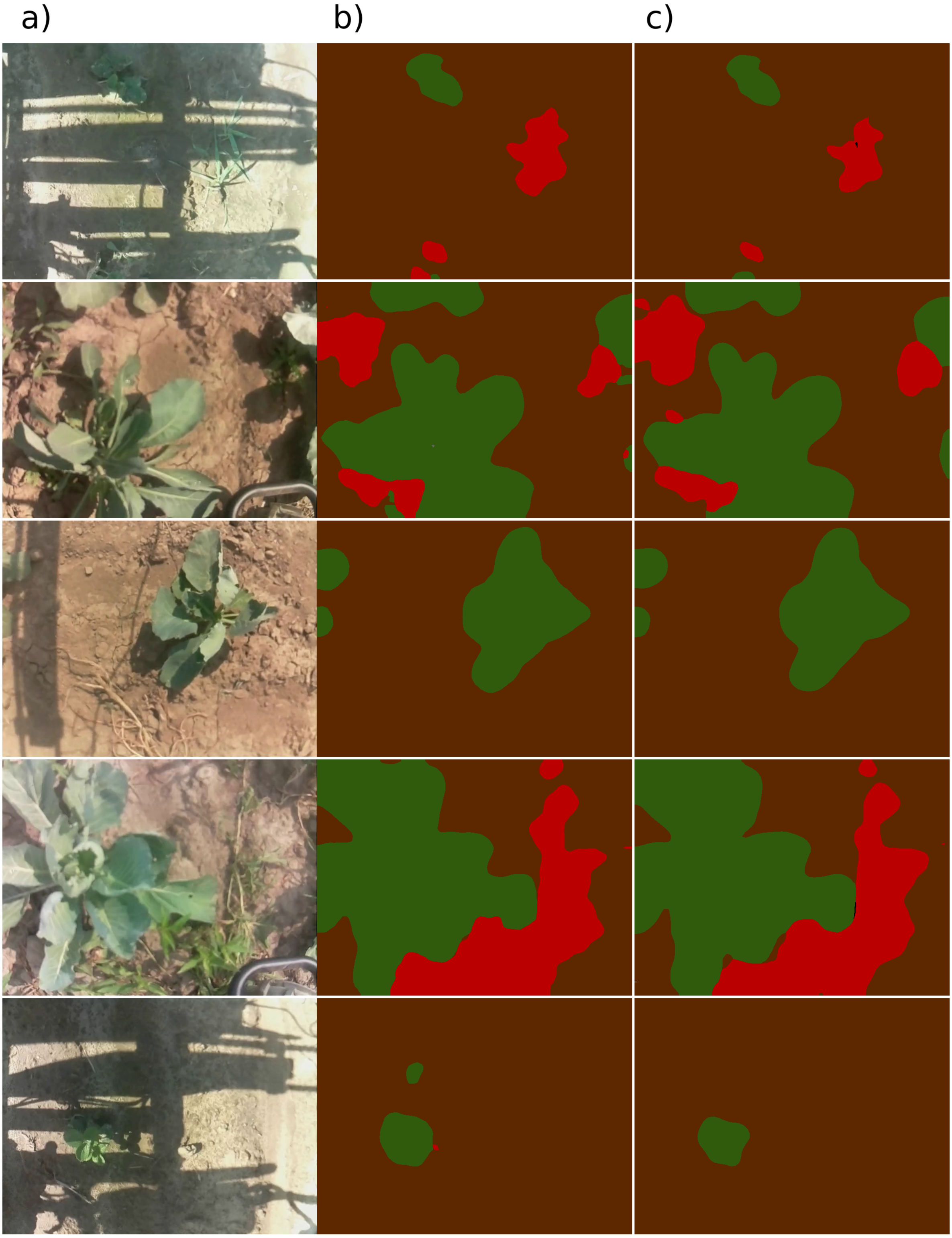

Figure 4. Example predictions of the stationary model: (a) input images (b) segmentation results for a single model instance (c) segmentation results for the ensemble of 50 model instances. Pixels with the highest probability for crops are marked green, weeds are marked red and the ground is marked brown.

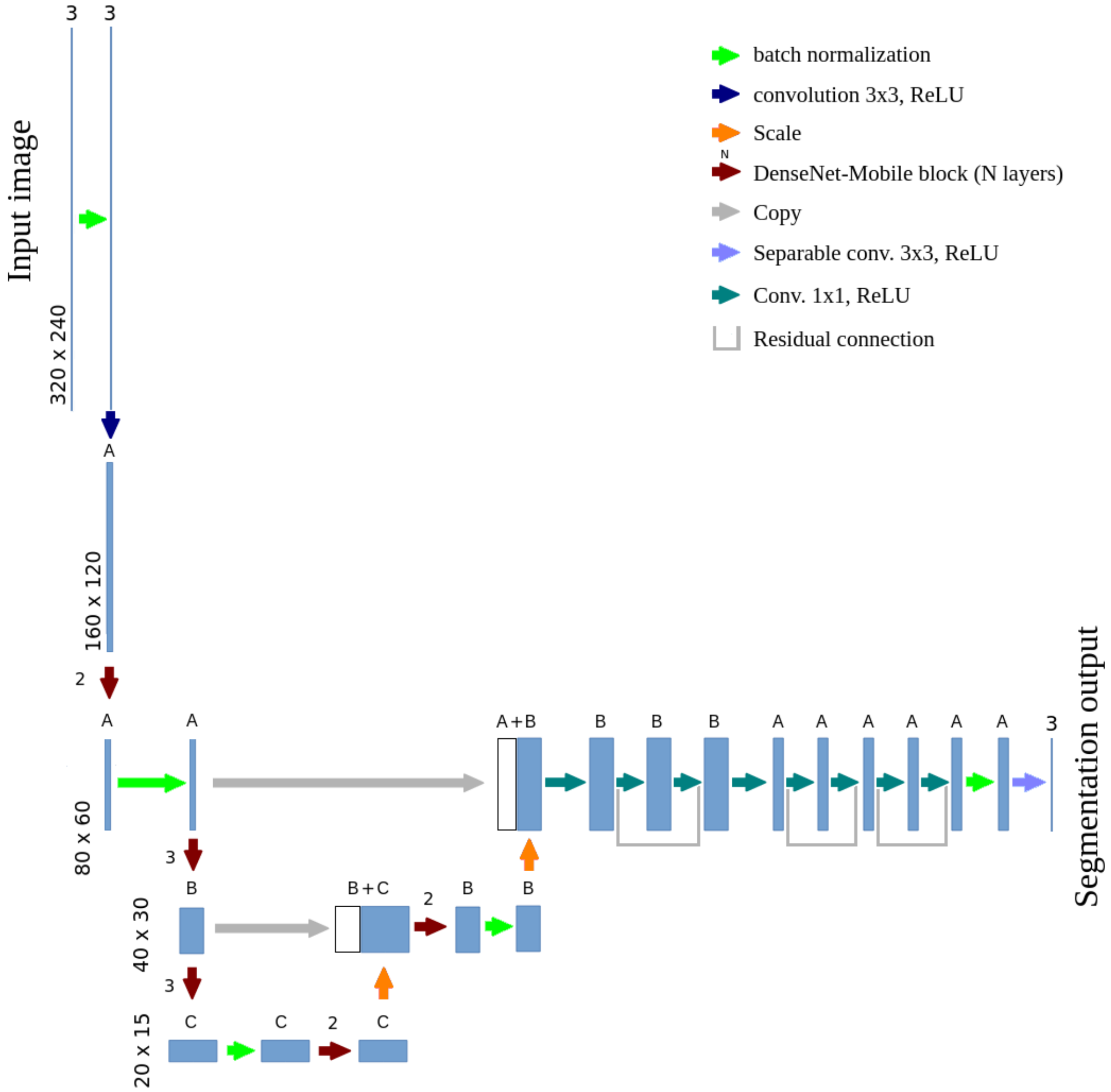

Figure 5. A scheme of the mobile CNN performing plant semantic segmentation, based on U-net, MobileNetsv2 and DenseNet architectures.

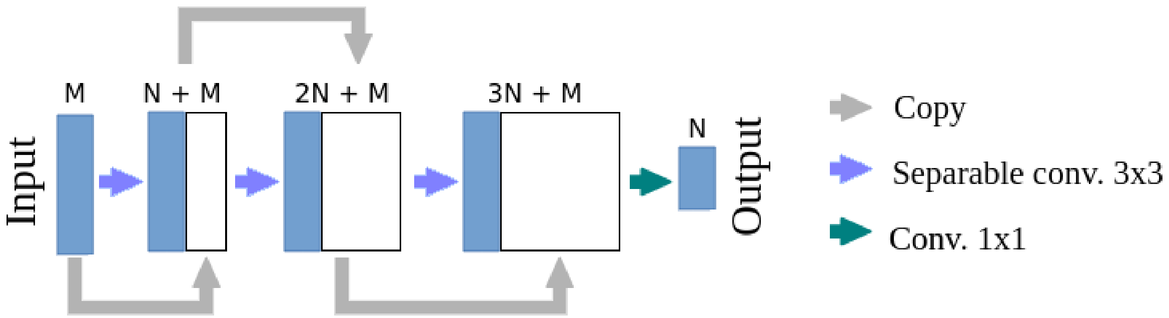

Figure 6. A structure of the DenseNet-Mobile block using separable convolution – naive implementation. The presented block contains three depthwise convolution layers and reduces the input resolution by a factor of two.

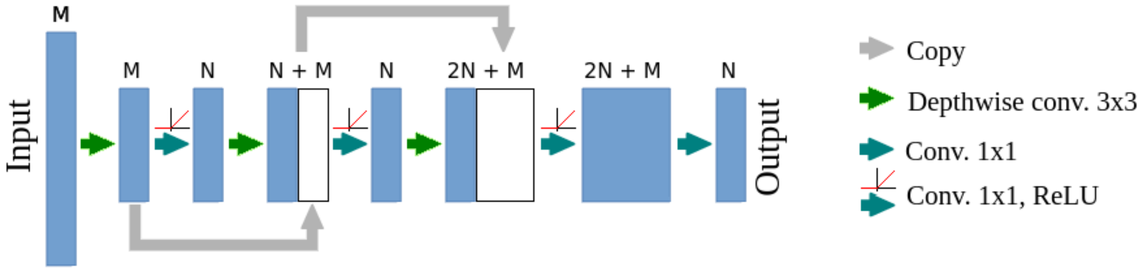

Figure 7. A structure of the DenseNet-Mobile block using separable convolution – custom implementation, reducing inference time. The presented block contains three depthwise convolution layers and reduces the input resolution by a factor of two.

chosen experimentally, as the smallest one allowing for monotonic inference time in function of the parameter α. In the custom version, the number of channels in each layer is defined as:

$$\begin{aligned} f_{mod8}(x) &= min(8, 8 \cdot \left\lfloor \frac{x}{8} \cdot \alpha \right\rceil) \\ A &= f_{mod8}(16) \\ B &= f_{mod8}(32) \\ C &= f_{mod8}(64) \end{aligned} \tag{2}$$

Segmentation error is defined as 100% minus average (for all C classes and N test images) value of the Dice coefficient, which equals to a quotient of a doubled area of a common part of the label E and the prediction P over the sum of them:

$$e = 100\% - \frac{1}{N \cdot C} \sum_{i=1}^{N} \sum_{j=1}^{C} \frac{2|E \cap P|}{|E| + |P|} \tag{3}$$

4. Experimental results

Inference time and segmentation error for different α values and four architecture versions is presented in figure 8. Sample prediction results are presented in figure 9. Note that segmentation error was also measured for the best configuration (with the custom parameter α and DenseNet-Mobile block), but with direct training on manual labels, without the use of knowledge distillation technique. In such case for $\alpha = 1.0$ the average segmentation error equals $18.3 - 19.4\%$ (training was repeated twice), which is out of range of figure 8. The custom modification of the α parameter definition resulted in functional dependency between inference time and segmentation accuracy, while the custom structure of a DenseNet-Mobile block resulted in a significant reduction of inference time.

Please note that segmentation error is based on the difference between a mobile model and the ensemble of stationary models, due to difficulties in obtaining a large database of high-quality manual labels. However, the selection of the final model will require an expert's decision even if such a database is present because different kind of segmentation errors differently influences the automatic weeding process. Final user-oriented error metric should include all gains and losses (e.g. missed weeds, destroyed crops in case of mechanical weeding, amount of herbicides used in case of their precise usage).

5. Conclusions

Described novelties allowed to reach satisfying accuracy at inference time which enables efficient control of the weeding machine (50 – 100 ms per frame on Raspberry Pi). Intensive tests will be required while the final mechanical structure of the device will be developed (which is planned in

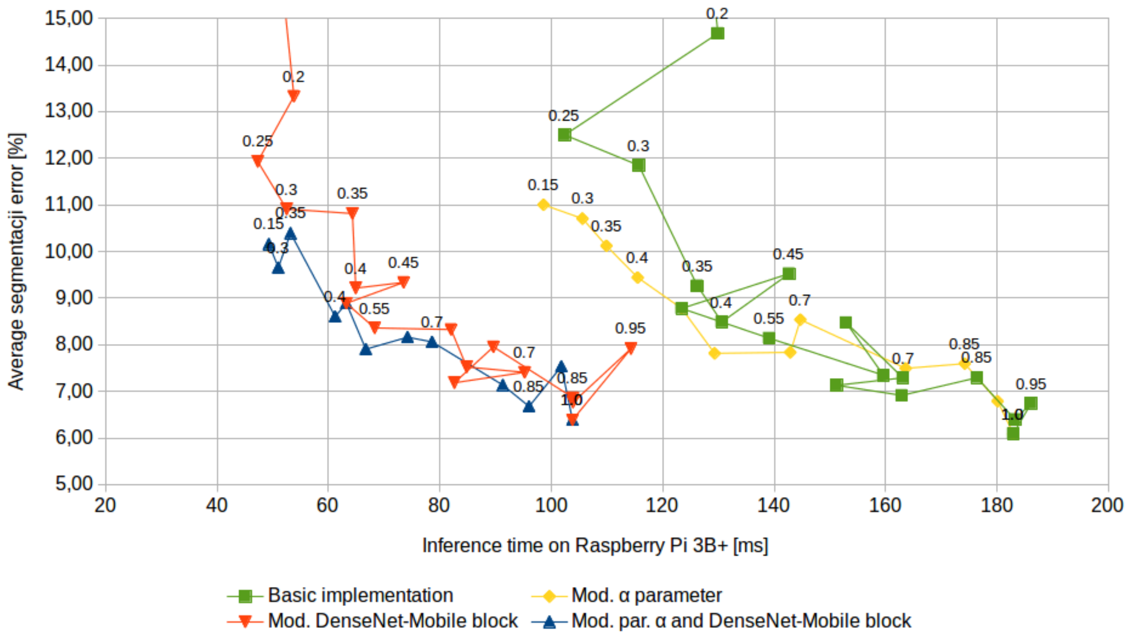

Figure 8. Segmentation error and inference time for different values of α (denoted above selected points to avoid crowd). Measurement points connected with lines to mark order of α changes. Without the custom definition of the α parameter, the dependency between segmentation error and inference time is nonfunctional. Without the custom version of the DenseNet-Mobile block inference time is significantly higher.

2020), including different plant species, vegetation levels, and weather conditions. In case of difficulties in obtaining satisfying accuracy in the final conditions the following remedies can be used:

- Training of different models for different plant species and growth stadiums and additional model for crops recognition used once at the weeding start
- Change of the α parameter value
- Change of the number of final residual layers
- Reduction of the number of batch normalization operations.

The remedies were proposed according to the risk management rules for research projects. However, in the authors' opinion, the probability of having to use them is low and the obtained results introduce novelties making the weeding machine innovative.

Author Contributions: Conceptualization, Łukasz Chechliński and Barbara Siemiątkowska; Data curation, Łukasz Chechliński and Michał Majewski; Formal analysis, Łukasz Chechliński and Barbara Siemiątkowska; Methodology, Łukasz Chechliński and Barbara Siemiątkowska; Software, Łukasz Chechliński; Validation, Łukasz Chechliński and Michał Majewski; Visualization, Łukasz Chechliński; Writing – original draft, Łukasz Chechliński and Barbara Siemiątkowska; Writing – review & editing, Łukasz Chechliński and Barbara Siemiątkowska.

Funding: This research was partially funded by Polish NCBiR grant number POIR.01.01.01-00-0974/16 and partially by the statutory resources.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

1. Chechliński, Ł.; Siemiątkowska, B.; Majewski, M. A System for Weeds and Crops Identification Based on Convolutional Neural Network. Conference on Automation. Springer, 2018, pp. 193–202.

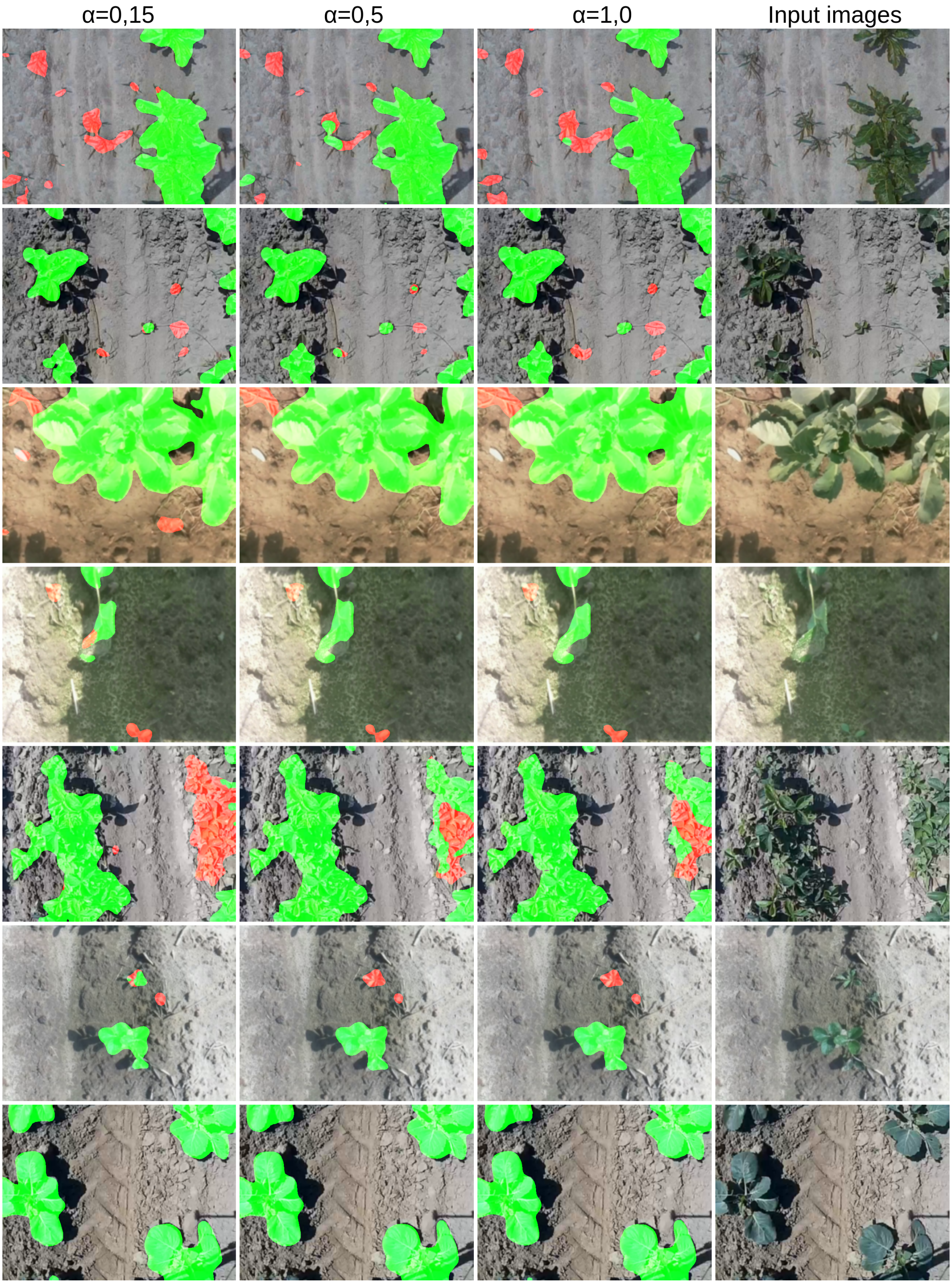

Figure 9. Example prediction results of the mobile model with a custom structure of the DenseNet-Mobile block and a custom definition of the α parameter. In columns images with highlighted prediction results for different values of the α parameter, the input image in the right-most column. Crops are highlighted green and weeds are highlighted red.

2. Sardana, V.; Mahajan, G.; Jabran, K.; Chauhan, B.S. Role of competition in managing weeds: An introduction to the special issue. *Crop Protection* **2017**, *95*, 1 – 7. Role of crop competition in weed management, doi:https://doi.org/10.1016/j.cropro.2016.09.011.
3. Sardana, V.; Mahajan, G.; Jabran, K.; Chauhan, B.S. Role of competition in managing weeds: An introduction to the special issue. *Crop Protection* **2017**, *95*, 1–7.
4. Mystkowska, I.; Zarzecka, K.; Baranowska, A.; Gugała, M.; others. An effect of herbicides and their mixtures on potato yielding and efficacy in potato crop. *Progress in Plant Protection* **2017**, *57*, 21–26.
5. Sujaritha, M.; Lakshminarasimhan, M.; Fernandez, C.J.; Chandran, M. Greenbot: a solar autonomous robot to uproot weeds in a grape field. *International Journal of Computer Science and Engineering* **2016**, *4*, 1351–1358.
6. Vigilant, V.; Rover, I.E. VVINNER: an autonomous robot for automated scoring of vineyards.
7. Wäldchen, J.; Mäder, P. Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review. *Archives of Computational Methods in Engineering* **2017**. doi:10.1007/s11831-016-9206-z.
8. Priyankara, H.A.C.; Withanage, D.K. Computer assisted plant identification system for Android. 2015 Moratuwa Engineering Research Conference (MERCon), 2015, pp. 148–153. doi:10.1109/MERCon.2015.7112336.
9. Lowe, D.G. Object Recognition from Local Scale-Invariant Features. Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2; IEEE Computer Society: Washington, DC, USA, 1999; ICCV ’99, pp. 1150–.
10. Beghin, T.; Cope, J.S.; Remagnino, P.; Barman, S., Shape and Texture Based Plant Leaf Classification. In *Advanced Concepts for Intelligent Vision Systems: 12th International Conference, ACIVS 2010, Sydney, Australia, December 13-16, 2010, Proceedings, Part II*; Springer Berlin Heidelberg: Berlin, Heidelberg, 2010; pp. 345–353. doi:10.1007/978-3-642-17691-3_32.
11. Wu, S.G.; Bao, F.S.; Xu, E.Y.; Wang, Y.; Chang, Y.; Xiang, Q. A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network. *CoRR* **2007**, *abs/0707.4289*.
12. Tsolakidis, D.G.; Kosmopoulos, D.I.; Papadourakis, G., Plant Leaf Recognition Using Zernike Moments and Histogram of Oriented Gradients. In *Artificial Intelligence: Methods and Applications: 8th Hellenic Conference on AI, SETN 2014, Ioannina, Greece, May 15-17, 2014. Proceedings*; Springer International Publishing: Cham, 2014; pp. 406–417. doi:10.1007/978-3-319-07064-3_33.
13. Caglayan, A.; Guclu, O.; Can, A.B., A Plant Recognition Approach Using Shape and Color Features in Leaf Images. In *Image Analysis and Processing – ICIAP 2013: 17th International Conference, Naples, Italy, September 9-13, 2013, Proceedings, Part II*; Springer Berlin Heidelberg: Berlin, Heidelberg, 2013; pp. 161–170. doi:10.1007/978-3-642-41184-7_17.
14. Yanikoglu, B.; Aptoula, E.; Tirkaz, C. Automatic Plant Identification from Photographs. *Mach. Vision Appl.* **2014**, *25*, 1369–1383. doi:10.1007/s00138-014-0612-7.
15. Zhang, J.; He, L.; Karkee, M.; Zhang, Q.; Zhang, X.; Gao, Z. Branch detection for apple trees trained in fruiting wall architecture using depth features and Regions-Convolutional Neural Network (R-CNN). *Computers and Electronics in Agriculture* **2018**, *155*, 386 – 393. doi:https://doi.org/10.1016/j.compag.2018.10.029.
16. Chavan, T.R.; Nandedkar, A.V. AgroAVNET for crops and weeds classification: A step forward in automatic farming. *Computers and Electronics in Agriculture* **2018**, *154*, 361–372.
17. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* **2017**.
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
19. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.
20. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. International Conference on Medical image computing and computer-assisted intervention. Springer, 2015, pp. 234–241.
21. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2881–2890.