

Article

Lightweight Semantic Segmentation Network for Real-Time Weed Mapping Using Unmanned Aerial Vehicles

Jizhong Deng ^{1,2}, Zhaoji Zhong ^{1,2}, Huasheng Huang ³ , Yubin Lan ^{2,4}, Yuxing Han ^{2,4,*} and Yali Zhang ^{1,2,*} 

¹ College of Engineering, South China Agricultural University, Wushan Road, Guangzhou 510642, China; jz-deng@scau.edu.cn (J.D.); zhongzhaoji@stu.scau.edu.cn (Z.Z.)

² National Center for International Collaboration Research on Precision Agricultural Aviation Pesticide Spraying Technology, Wushan Road, Guangzhou 510642, China; ylan@scau.edu.cn

³ College of Computer Sciences, Guangdong Polytechnic Normal University, Zhongshan Road, Guangzhou 510642, China; huanghsheng@126.com

⁴ College of Electronic Engineering, South China Agricultural University, Wushan Road, Guangzhou 510642, China

* Correspondence: yuxinghan@scau.edu.cn (Y.H.); ylzhang@scau.edu.cn (Y.Z.)

Received: 19 September 2020; Accepted: 12 October 2020; Published: 13 October 2020



Abstract: The timely and efficient generation of weed maps is essential for weed control tasks and precise spraying applications. Based on the general concept of site-specific weed management (SSWM), many researchers have used unmanned aerial vehicle (UAV) remote sensing technology to monitor weed distributions, which can provide decision support information for precision spraying. However, image processing is mainly conducted offline, as the time gap between image collection and spraying significantly limits the applications of SSWM. In this study, we conducted real-time image processing onboard a UAV to reduce the time gap between image collection and herbicide treatment. First, we established a hardware environment for real-time image processing that integrates map visualization, flight control, image collection, and real-time image processing onboard a UAV based on secondary development. Second, we exploited the proposed model design to develop a lightweight network architecture for weed mapping tasks. The proposed network architecture was evaluated and compared with mainstream semantic segmentation models. Results demonstrate that the proposed network outperform contemporary networks in terms of efficiency with competitive accuracy. We also conducted optimization during the inference process. Precision calibration was applied to both the desktop and embedded devices and the precision was reduced from FP32 to FP16. Experimental results demonstrate that this precision calibration further improves inference speed while maintaining reasonable accuracy. Our modified network architecture achieved an accuracy of 80.9% on the testing samples and its inference speed was 4.5 fps on a Jetson TX2 module (Nvidia Corporation, Santa Clara, CA, USA), which demonstrates its potential for practical agricultural monitoring and precise spraying applications.

Keywords: semantic segmentation network; UAV; real time; embedded device; weed mapping; deep learning

1. Introduction

In farmland, timely weed control is crucial for ensuring optimal crop production. To minimize the negative effects of herbicides, site-specific weed management (SSWM) protocols [1] are essential. For SSWM, farmers obtain orthophoto maps of their fields, formulate an application plan, and generate

prescription maps. Methods for obtaining orthophoto maps include satellite imaging and UAV imaging [2]. The resulting remote sensing images are analyzed to map weed distribution information, which can provide decision-making support information for precise spraying applications [3].

Based on their high spatial resolution, UAVs have been found to be appropriate for weed mapping tasks. Ortizet et al. [4] collected UAV images and achieved good performance for weed mapping using a semi-supervised method. Castaldi et al. [5] used UAV multispectral images to classify maize and weeds. However, in previous studies, most image processing has been conducted offline, which results in a significant time gap between image collection and spraying applications. This disadvantage significantly limits the applications of SSWM because agronomy conditions may change during large time gaps. In SSWM applications, the time gaps should be less than one day so that the herbicide treatment can be applied before the agronomy conditions change. One possible solution to address this issue is to utilize real-time image processing onboard UAVs. Based on real-time data collection performed by embedded devices, UAVs can perform corresponding spraying actions immediately, which can solve the problem of time gaps in the current model of SSWM.

Although there has been no related work in the domain of SSWM research, we identified some related studies in industrial domains. One of the barriers in real-time image processing is that the image data is large and difficult to process by embedded devices, and some studies have exploited to reduce the amount of data while maintaining the monitoring objective. Barmpoutis et al. [6] used a 360-degree camera mounted on a UAV for fire detection. The use of 360-degree camera significantly reduces the amount of data, which is meaningful in real-time processing scenarios. In addition to the data catastrophe, the employment of deep learning models utilized by connecting desktops to embedded devices is a major challenge in the industrialization of artificial intelligence. Currently, most studies on employing deep learning models using embedded devices focus on reducing the model size while maintaining reasonable accuracy. Alexander et al. [7] introduced a tiny solid-state drive architecture that requires a storage space of only 2.3 MB to achieve a mean average precision of 61.3% on the VOC 2007 challenge. Sabir et al. [8] proposed a lightweight model for multiple-object detection and tracking that can integrate a deep-learning-based association metric approach with simple online and real-time tracking. Their algorithm was deployed on an NVIDIA Jetson TX module and Intel Neural Compute Stick. The results demonstrated the effectiveness of their algorithms for real-time experiments onboard UAVs. Arpit et al. [9] proposed a modified MobileNetV2 architecture for fire classification. They modified the last fully-connected layer of the MobileNetV2 architecture to fit the fire detection task, and the resulting model exhibited better performance than that of previous models. The proposed model was deployed on a Raspberry Pi 3B device and achieved a speed of 5 fps with an accuracy of 0.92 on the dataset presented in [10].

However, regarding the requirements of SSWM applications, there are some limitations to the methods proposed in the aforementioned studies. (1) Although previous studies have developed various approaches to reduce the number of model parameters, they have largely focused on classification [9] and detection models [7,8,11]. In studies on SSWM, semantic segmentation models [12] have proven to be effective at weed mapping tasks [13,14]; however, their optimization cannot be guaranteed based on the results of the aforementioned studies. One of the greatest obstacles for implementing such deep learning networks for widespread deployment on embedded devices is their high computational and memory requirements. In general, semantic segmentation models are difficult to optimize for small devices because they use 2D information that requires significant computation, making them less suitable for the research described in this paper. (2) The aforementioned studies only considered model design and optimization, while ignoring the optimization of the inference process, which may have the potential to achieve better efficiency. (3) Most previous studies have only conducted model design and validation on computers or embedded devices. However, in the context of SSWM applications, a real-time spraying machine requires modules for map visualization, flight control, image collection, image processing, and spraying. Based on the objectives of this study, it was

necessary to construct a basic hardware environment to support modules for map visualization, route planning, image collection, and image processing.

Therefore, the main goals of this study were to (1) develop a hardware system integrating map visualization, flight control, image collection, and image processing; (2) perform model design to achieve a compact size while maintaining reasonable prediction accuracy; and (3) exploit the optimization of the inference process, which can further improve inference speed. There are two main contributions of this work. First, we propose a lightweight network architecture for weed mapping tasks, which improves inference speed while maintaining reasonable accuracy. Second, we propose combining precision calibration with model design to improve speed, which has not been discussed in other related papers.

2. Hardware Development for a UAV System

The developed system can be divided into three main components: (1) a map visualization module for selecting flight areas and generating flight plans, (2) flight control module for controlling the UAV, and (3) onboard image collection and processing module. These components are illustrated in Figure 1a. To apply this system in an experiment, a user must first select an area of interest on a map. The software on the laptop then automatically generates corresponding routes. Waypoints are then uploaded to the flight control system via wireless serial communication. The flight control system controls the UAV and begins to collect images. During the flight process, newly collected images are processed directly by the image processing module. In this step, a fully convolutional network (FCN) is used to map weed distributions at the pixel level. Finally, a weed cover map is generated in real-time, which can provide decision support information for spraying machines. In this study, we adopted the M100 UAV platform (DJI Co., Ltd., Shenzhen, China). The Jetson TX2 module (Nvidia Corporation, Santa Clara, CA, USA) was selected as a control center to realize flight control, image collection, and image processing. The main components of the UAV can be observed in Figure 1b. The M100 UAV platform was chosen for its flight stability against airflow disturbance, and the Jetson TX2 was chosen for its GPU unit which is powerful in the inference of the deep learning models.



Figure 1. Composition of the developed system. (a) Components of the developed system. (b) Main components of the UAV system.

2.1. Map Visualization Module

In this study, the map visualization module (see Figure 2a) was developed using the Google Maps API (Google Inc., Mountain View, CA, USA). The ground station software has two main functions: (i) selecting areas of interest on the map and automatically generating routes, and (ii) transferring flight points to the flight control system via serial communication. The main objective of the ground station software is to provide a map for a user to select an area of interest. The selection of an area of interest is a basic requirement for the subsequent route planning procedure, autonomous flight, and image acquisition performed by the UAV. The map used in this software is a mosaic of map tiles downloaded from Google Maps. Map tiles are offline resources on the laptop that do not need to be accessed via

the internet. The map only provides a real-world overview of the area and does not contain specific location information, such as building locations, obstacles, and real-time weather data. Before flight planning begins, the user must define the forward lap and side lap. The overlap rate, flight altitude, and resolution of the camera are then used to generate flight routes. As shown in Figure 2b, during the course of a flight mission, UAVs perform image acquisition along the planned waveform path.

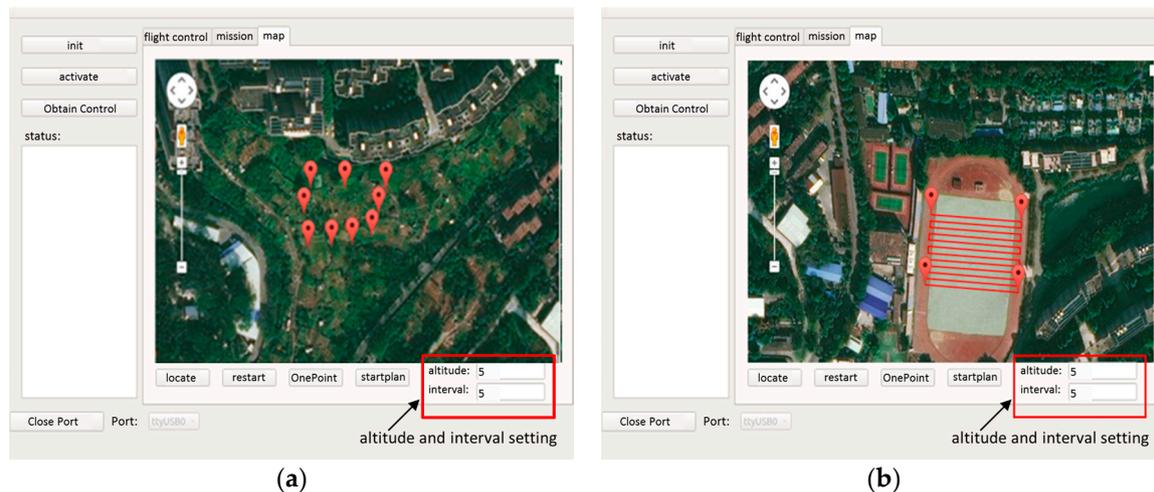


Figure 2. The map module. (a) User interface of the map module. (b) One example of route planning.

2.2. Flight Control Module

In this study, the flight control system was developed using the open-source Onboard Software Development Kit provided by the producer of the M100 (DJI Co., Ltd., Shenzhen, China). The flight control system realizes secondary control of the UAV platform. Communication between the flight control system and laptop is realized using a wireless serial module. Communication between the flight control system and UAV platform is realized using the controller area network protocol [15].

2.3. Image Collection and Processing

The image processing module in the UAV has two main functions: (i) automatically directing image acquisition according to the planned routes and predefined overlaps and (ii) processing collected images in real time. The camera (see Figure 1b) used in the image acquisition module of the system is a TOP-T10X camera (Tuopu Lianchuang co., Ltd., Beijing, China) [16], which is stabilized by an optimized three-axis gimbal. On the M100 UAV, image acquisition is accomplished using the “opencvsdk” library [17]. Figure 3 presents the overall workflow of the system. The software acquires the most recent frame from the camera at a resolution of 1280×720 pixels. Images are cropped according to the central 720×720 pixel area and then resized to 1000×1000 pixels, which was proven to be a suitable image size for weed mapping by the previous work of our team [14]. The camera can acquire images at up to 60 fps. The overall processing speed is determined by the network model and the capabilities of the GPU.

2.4. Task Assignment

During flight, the Jetson TX2 module performs several tasks. It is necessary to assign these tasks to different components onboard the Jetson TX2 module. It is equipped with a 256 core NVIDIA Pascal GPU and six core ARMv8 64-bit CPU complex. According to the designs presented in related papers, the CPU serves as the host and is responsible for task scheduling, while the GPU serves as a device and performs the computing tasks required for image processing. For semantic segmentation models, more than 80% of the operation time of most algorithms is consumed by convolution, deconvolution, and matrix operations. A GPU can handle these high-density and high-volume operations using

highly parallel multi-threading, while the CPU is responsible for the UAV flight control program and communication processing, as shown in Figure 4.

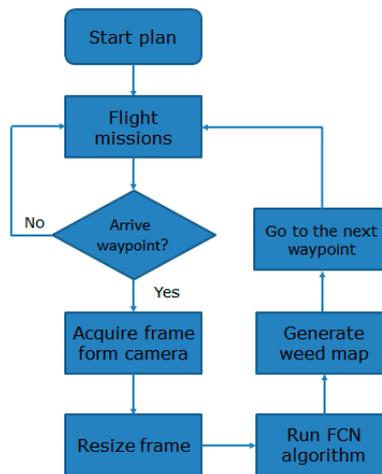


Figure 3. Overall workflow for image collection and processing.

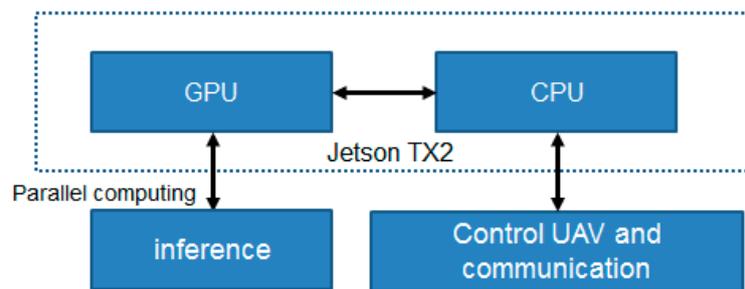


Figure 4. Task assignment on the Jetson TX2 module.

3. Model Design and Optimization

3.1. Model Design

Related studies have demonstrated that a FCN is appropriate for weed mapping tasks. In contrast to a traditional convolutional neural network, an FCN is an end-to-end network structure that can perform dense prediction tasks, allowing it to determine the weed densities at a target site, which can be directly used as references for accurate spraying tasks. For an FCN, input data can be images of arbitrary size and a corresponding output size can be obtained through effective inference. An FCN transforms all fully-connected layers into convolutional layers. In this form, the spatial information of an input image is preserved. By using deconvolutional operations, the feature maps screened from the final convolutional layer are up-sampled and the output image is restored to the same size as the input image. The general architecture of an FCN is presented in Figure 5.

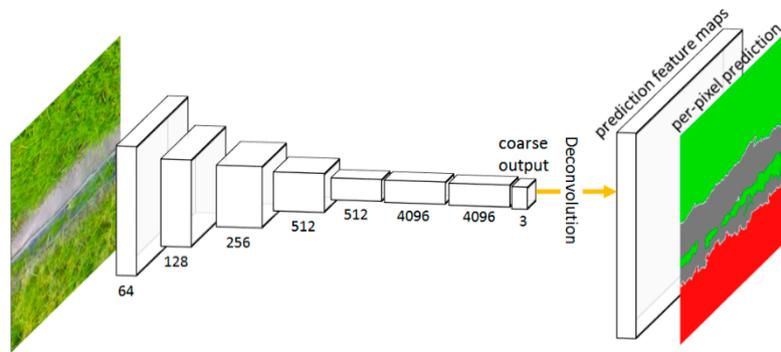


Figure 5. General architecture of an FCN.

Similar to other studies, we adopted a pre-trained network and fine-tuned it using a relevant dataset. Commonly used pre-trained networks include the ResNet [18] and VGGNet [19] models. These models tend to outperform their counterparts in terms of accuracy. However, according to the research presented in [14], these models are relatively time consuming, that is, they are inappropriate for real-time tasks. Therefore, we selected AlexNet as a pre-trained model because it is a relatively lightweight framework. As shown in Figure 6, AlexNet contains five convolutional layers and three fully connected layers. To reduce the number of model parameters, we removed the FC6 and FC7 layers. Additionally, the number of neurons in the final fully connected layer was reduced to three to match the number of classes (each pixel was classified as “rice”, “weed”, or “other”). Next, the last fully connected layer was transformed into a convolutional layer and a deconvolutional layer was appended to restore the feature map to the original image resolution.

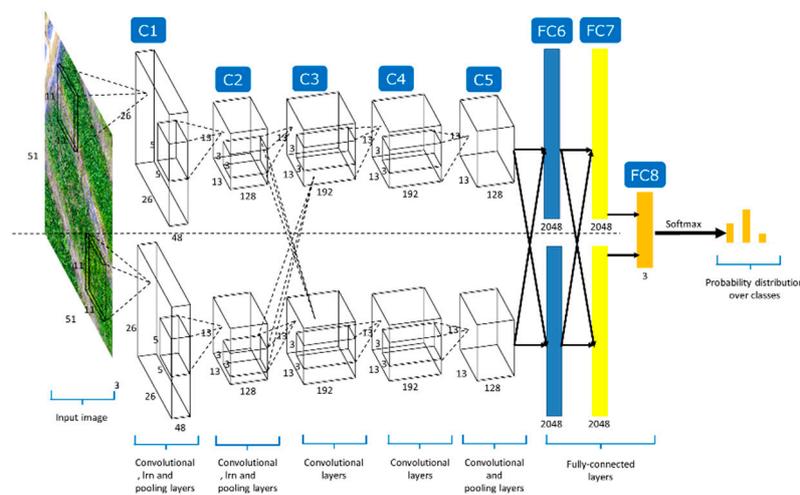


Figure 6. General architecture of AlexNet.

3.2. Optimization of the Inference Process

Most deep learning models train and evaluate neural networks with full 32-bit precision (FP32). However, once a model is fully trained, inference computations can use half-precision FP16 instead because gradient backpropagation is not required for inference. Using a lower precision results in a smaller model size, lower memory utilization and latency, and higher throughput. In this case, we applied FP32 for training and used FP16 for inference. However, it is worth noting that this will have an impact on recognition accuracy.

4. Results

In this section, we proposed a light weight network for weed mapping and compared its performance with the mainstream segmentation models. After that, the proposed network was transferred from the desktop to the embedded devices. Precision calibration was used to optimize the inference process. All models were evaluated in terms of accuracy and efficiency. For the accuracy, the metrics of overall accuracy and mean intersection over union (IoU) were applied. The overall accuracy is computed using the ratio of the pixels accurately classified to all pixels, and the mean intersection over union is calculated using the percent overlap between the target mask and our prediction output. For the efficiency, the frames per second (fps) was used as the metric, which represents how many images can be processed in one second.

4.1. Data Collection

In this study, experimental data were collected from a rice field located in Southern China (113.636888 N, 23.240441 E). Data collection was conducted when the weeds and crops were in their early tillering stages. The flight height was set to 6 m above the ground and the forward lap and side lap for imaging were set to 50% and 60%, respectively. A total of 1092 samples were collected and each sample was a 1000 × 1000 pixel RGB UAV image, where each image corresponds to a 30 cm × 30 cm field area. Each pixel in each image was classified as one of three classes: rice, weed, and other. To perform network training and validation, the images were manually labeled at the pixel level, as shown in Figure 7. In our dataset, 892 samples were used for training and the remaining 200 images were used for network validation.

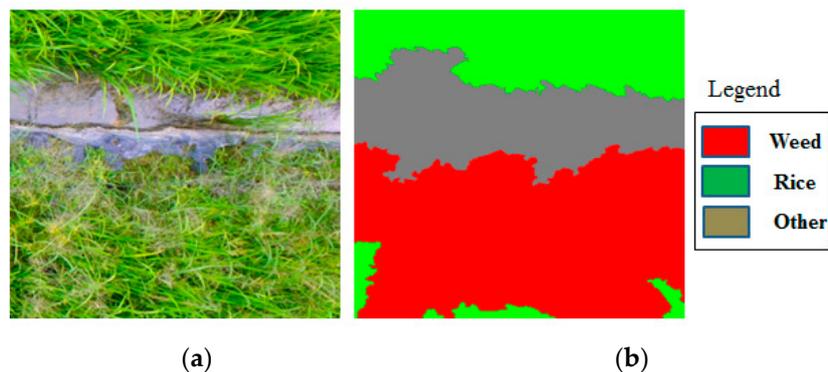


Figure 7. Example of image labeling: (a) Input images, and (b) labels.

4.2. Model Design

As described in Section 3.2., a lightweight classification architecture (AlexNet) was selected as a baseline network. To reduce the number of network parameters, the first two fully-connected layers were removed, the number of neurons of the final fully-connected layer was set to three, and this layer was eventually transformed into a convolutional layer. A deconvolutional layer was appended to restore the feature map to the original image resolution. The training data were augmented using data enhancement (translation, rotation, and tailoring).

During the training process, the Adam optimizer was used for adaptive momentum estimation. The Adam gradient descent method is an adaptive learning rate method with different parameters that yields a fast convergence speed. The entire training process was limited to 30 epochs, where each iteration traversed all training samples. After forward computation, the sigmoid cross-entropy loss function was used as a cost function. The initial learning rate was set to 0.00001. Since the learning rate can be adjusted adaptively according to the rate of gradient descent, the convergence speed of the model was improved, as shown in Figure 8. After 30 epochs, testing on the validation dataset resulted in an overall accuracy of 91.2%. As shown in Table 1, the model operates on a GTX1060 GPU at 0.9 fps,

while implementing the same network on the Jetson TX2 module results in a rate of only 0.11 fps, which is insufficient for real-time processing applications.

Our modified AlexNet-FCN was compared with the classic VGGNet, GoogLeNet, and ResNet. For the VGGNet-FCN, the skip architecture was omitted to increase speed. For the GoogLeNet, ReLU was used as the activation function, and the dropout strategy was used to overcome the problem of overfitting. For the ResNet-FCN, the ResNet-101 model was selected as a baseline architecture because it provides the best balance of accuracy and efficiency [20]. The performances of all models are summarized in Table 1. One can see that the accuracy of AlexNet-FCN is slightly lower than that of VGGNet-FCN and GoogLeNet-FCN. However, it is almost four times faster than the VGGNet-FCN and two times faster than the GoogLeNet-FCN, meaning it can meet the requirements of real-time scenarios. The accuracy of the ResNet-FCN is the highest among the three models based on its deep structure and residual architecture. However, this model is so large that the GTX 1060 GPU could not even perform the forward computation. In this case, we used the GTX 1080 TI GPU for network training and evaluation. However, the GPU resources of the GTX 1060 are greater than those of embedded devices. Therefore, it is impossible to copy the ResNet-FCN from the GTX 1080 TI GPU to the embedded Jetson TX2 module.

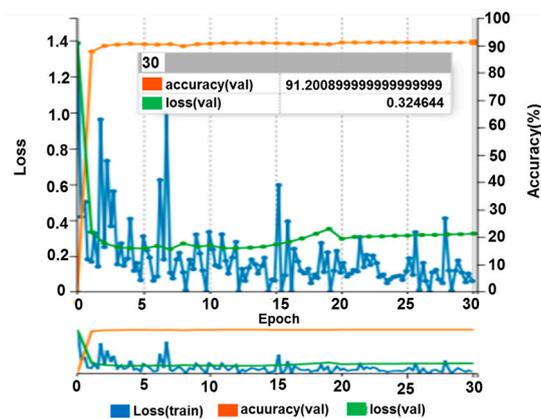


Figure 8. The loss and accuracy curves during training process.

Table 1. Comparisons of the proposed FCN-Alexnet with other mainstream network architectures.

Model	Device	Overall Accuracy (%)	Mean IoU (%)	Frames Per Second
The proposed FCN-Alexnet model	GTX 1060	91.2	70.5	12.5
VGGNet-FCN by Simonyan et al. [19]	GTX 1060	92.3	72.8	3.1
GoogLeNet-FCN by Szegedy et al. [21]	GTX 1060	91.9	71.3	5.9
ResNet-FCN by He et al. [18]	GTX 1080 TI	94.2	77.2	9.3

4.3. Optimization of the Inference Process

To reduce the model size, the data precision requirement of the network was changed from FP32 to FP16. The Jetson TX2 module uses half-precision mode to reduce the size of the model to half of its original size. Table 2 lists the results of precision calibration for the modified AlexNet-FCN. One can see that precision calibration improves inference efficiency by approximately three to four times on both the GTX 1060 (Nvidia Corporation, Santa Clara, CA, USA) and Jetson TX2 module. Although the prediction accuracy decreases slightly, the overall accuracy is acceptable for agricultural applications because it can provide rapid judgments regarding weed density.

Table 2. Inference optimization in the desktop and the embedded devices.

Device	Precision Clibration	Overall Accuracy (%)	Mean IoU (%)	Frames Per Second
GTX 1060	FP32	91.2	70.5	12.5
Jetson TX2	FP32	91.2	70.5	1.2
GTX 1060	FP16	80.9	62.8	35.6
Jetson TX2	FP16	80.9	62.8	4.5

Figure 9 presents the output images corresponding to several testing samples before and after precision calibration. In Figure 9c, one can see that our modified AlexNet-FCN can correctly distinguish the rice and weed areas. Following precision calibration, the prediction accuracy decreases by a small amount. The weed area in the third sample (blue dotted lines) was classified as the “other” category by the model following precision calibration. However, following precision calibration, our model still correctly classifies most areas in the input images, demonstrating its capability to provide decision support information for spraying machines.

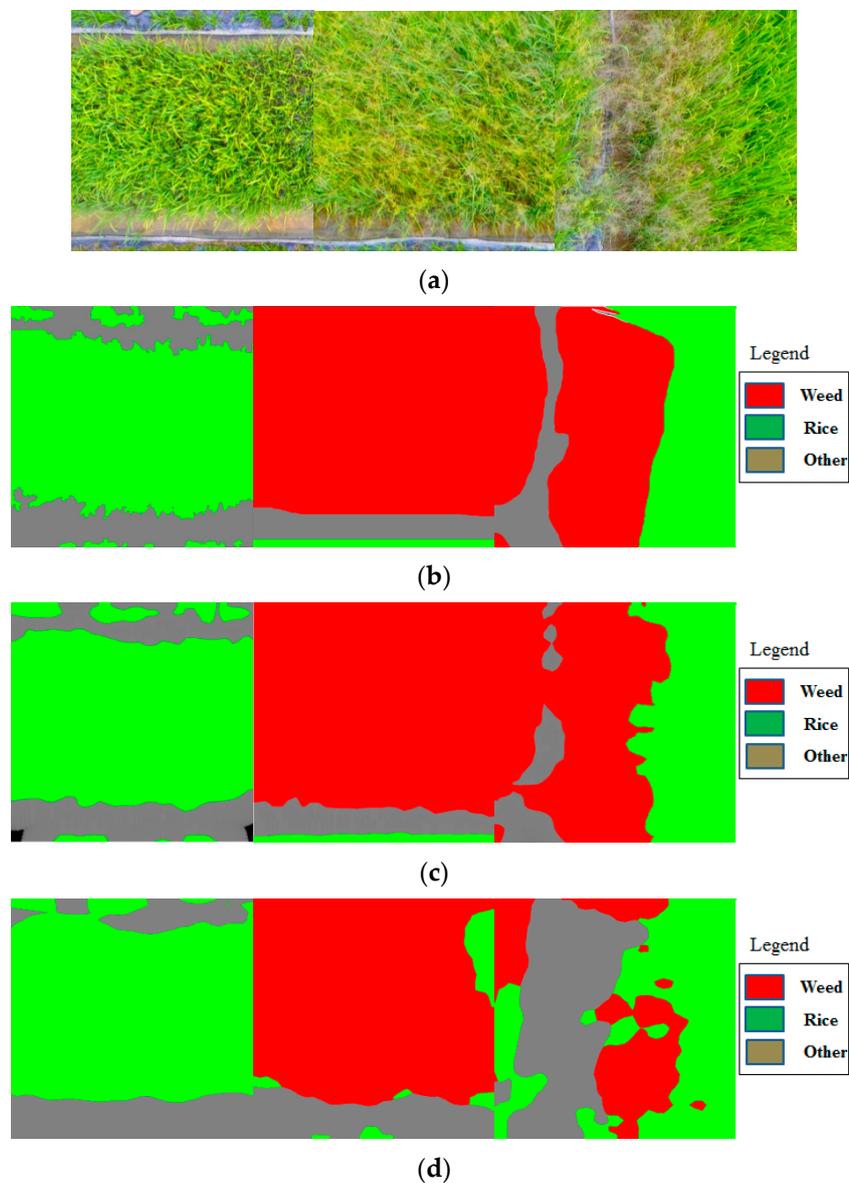


Figure 9. (a) UAV imagery, (b) corresponding labels, (c) results obtained by FCN-AlexNet, and (d) results obtained by FCN-AlexNet optimized with precision calibration.

5. Conclusions

UAV remote sensing and precision spraying are two important components of SSWM. However, the time gap between image collection and herbicide treatment significantly limits the application of SSWM technology. This study used real-time image processing onboard a UAV with the goal of eliminating the time gap between image collection and herbicide treatment. First, we established a hardware environment for real-time image processing that integrates map visualization, flight control, image collection, and real-time image processing onboard a UAV based on secondary development. Second, we exploited the proposed model design and presented a lightweight network architecture for weed mapping tasks. The proposed network architecture was evaluated and compared with mainstream semantic segmentation models. Experimental results demonstrated that the proposed network is almost two times faster than the mainstream segmentation models with competitive accuracy. Next, we conducted optimization of the inference process. Precision calibration was applied on both desktop and embedded devices and the precision was reduced from FP32 to FP16. Experimental results demonstrated that precision calibration further improves inference speed while maintaining reasonable prediction accuracy. Our modified network architecture achieved an accuracy of 80.9% on the testing samples and the inference speed of 4.5 fps on the Jetson TX2 module, which demonstrates its potential for practical agricultural monitoring and precise spraying applications.

In the future, we plan to collect additional UAV images for model training and validation. Additionally, we plan to combine our system with variable spraying technology. The classification results of UAV images can provide decision-making information for sprayers, which can help maintain pesticide effects while reducing the use of chemicals.

Author Contributions: Conceptualization: J.D., Y.H., and Y.Z.; funding acquisition: Y.L. and Y.Z.; methodology: Z.Z. and H.H.; project administration: J.D. and Y.Z.; software: Z.Z. and H.H.; writing—original draft: Z.Z.; writing—review and editing: J.D., H.H., and Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Key Area Research and Development Planning Project of Guangdong Province (grant no. 2019B020221001), Guangdong Provincial Innovation Team for General Key Technologies in Modern Agricultural Industry (grant no. 2019KJ133), the Science and Technology Planning Project of Guangdong Province, China (grant no. 2018A050506073), the National Natural Science Foundation of Guangdong Province, China (Grant No. 2018B030306026), the National Key Research and Development Plan, China (grant no. 2016YFD0200700), and the 111 Project, China (D18019).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. López-Granados, F. Weed detection for site-specific weed management: Mapping and real-time approaches. *Weed Res.* **2010**, *51*, 1–11. [[CrossRef](#)]
2. Wang, D.; Shao, Q.; Yue, H. Surveying wild animals from satellites, manned aircraft and unmanned aerial systems (UASs): A Review. *Remote Sens.* **2019**, *11*, 1308. [[CrossRef](#)]
3. Balafoutis, A.T.; Beck, B.; Fountas, S.; Vangeyte, J.; Van Der Wal, T.; Soto, I.; Gómez-Barbero, M.; Barnes, A.P.; Eory, V. Precision agriculture technologies positively contributing to GHG emissions mitigation, farm productivity and economics. *Sustainability* **2017**, *9*, 1339. [[CrossRef](#)]
4. Pérez-Ortiz, M.; Peña, J.; Gutiérrez, P.; Torres-Sánchez, J.; Hervás-Martínez, C.; López-Granados, F. A semi-supervised system for weed mapping in sunflower crops using unmanned aerial vehicles and a crop row detection method. *Appl. Soft Comput.* **2015**, *37*, 533–544. [[CrossRef](#)]
5. Castaldi, F.; Pelosi, F.; Pascucci, S.; Casa, R. Assessing the potential of images from unmanned aerial vehicles (UAV) to support herbicide patch spraying in maize. *Precis Agric.* **2016**, *18*, 76–94. [[CrossRef](#)]
6. Barmoutis, P.; Stathaki, T.; Dimitropoulos, K.; Grammalidis, N. Early fire detection based on aerial 360-degree sensors, deep convolution neural networks and exploitation of fire dynamic textures. *Remote Sens.* **2020**, *12*, 3177. [[CrossRef](#)]
7. Womg, A.; Shafiee, M.J.; Li, F.; Chwyl, B. Tiny SSD: A Tiny Single-Shot Detection Deep Convolutional Neural Network for Real-Time Embedded Object Detection. In Proceedings of the 2018 15th Conference

- on Computer and Robot Vision (CRV), Toronto, ON, Canada, 9–11 May 2018; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2018; pp. 95–101.
8. Hossain, S.; Lee, D.-J. Deep learning-based real-time multiple-object detection and tracking from aerial imagery via a flying robot with GPU-based embedded devices. *Sensors* **2019**, *19*, 3371. [[CrossRef](#)] [[PubMed](#)]
 9. Jadon, A.; Varshney, A.; Ansari, M.S. Low-complexity high-performance deep learning model for real-time low-cost embedded fire detection systems. *Procedia Comput. Sci.* **2020**, *171*, 418–426. [[CrossRef](#)]
 10. Foggia, P.; Saggese, A.; Vento, M. Real-time fire detection for video-surveillance applications using a combination of experts based on color, shape and motion. *IEEE Trans. Circuits Syst. Video Technol.* **2015**, *25*, 1545–1556. [[CrossRef](#)]
 11. Chen, S.; Lin, W. Embedded System Real-Time Vehicle Detection Based on Improved YOLO Network. In Proceedings of the 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, 11–13 October 2019; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2019; pp. 1400–1403.
 12. Fu, G.; Liu, C.; Zhou, R.; Sun, T.; Zhang, Q. Classification for high resolution remote sensing imagery using a fully convolutional network. *Remote. Sens.* **2017**, *9*, 498. [[CrossRef](#)]
 13. Huang, H.; Deng, J.; Lan, Y.; Yang, A.; Deng, X.; Zhang, L. A fully convolutional network for weed mapping of unmanned aerial vehicle (UAV) imagery. *PLoS ONE* **2018**, *13*, e0196302. [[CrossRef](#)] [[PubMed](#)]
 14. Huang, H.; Lan, Y.; Yang, A.; Zhang, Y.; Wen, S.; Deng, J. Deep learning versus object-based image analysis (OBIA) in weed mapping of UAV imagery. *Int. J. Remote. Sens.* **2020**, *41*, 3446–3479. [[CrossRef](#)]
 15. Kvaser CAN Protocol Tutorial. Available online: <https://www.kvaser.com/can-protocol-tutorial/> (accessed on 18 September 2020).
 16. TOP Pod. Available online: <http://www.topotek.com/typo-en.html> (accessed on 18 September 2020).
 17. Culjak, I.; Abram, D.; Pribanic, T.; Dzapov, H.; Cifrek, M. A Brief Introduction to OpenCV. In Proceedings of the 35th International Convention MIPRO, Opatija, Croatia, 21–25 May 2012; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2012; pp. 2142–2147.
 18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2016; pp. 770–778.
 19. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015; Available online: <https://arxiv.org/pdf/1409.1556.pdf> (accessed on 13 October 2020).
 20. Huang, H.; Lan, Y.; Deng, J.; Yang, A.; Deng, X.; Zhang, L.; Wen, S. A Semantic labeling approach for accurate weed mapping of high resolution UAV imagery. *Sensors* **2018**, *18*, 2113. [[CrossRef](#)] [[PubMed](#)]
 21. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2015; pp. 1–9.

