Journal of Agricultural Engineering

https://www.agroengineering.org/

Leveraging deep semantic segmentation for assisted weed detection

Francisco Garibaldi-Márquez,[1] Gerardo Flores,[2] Luis M. Valentín-Coronado[3,4]

[1]National Institute of Forestry, Agricultural and Livestock Research - Pabellón Experimental Field, Arteaga Pavilion, Aguascalientes, Mexico
[2]RAPTOR Lab, School of Engineering, College of Arts and Sciences, Texas A&M International University, Laredo, TX, USA
[3]Research Center in Optics, Leon, Guanajuato, Mexico
[4]National Council for Science and Technology (CONACyT), Mexico City, Mexico

Publisher's Disclaimer

E-publishing ahead of print is increasingly important for the rapid dissemination of science. The *Early Access* service lets users access peer-reviewed articles well before print/regular issue publication, significantly reducing the time it takes for critical findings to reach the research community.

These articles are searchable and citable by their DOI (Digital Object Identifier).

Our Journal is, therefore, e-publishing PDF files of an early version of manuscripts that undergone a regular peer review and have been accepted for publication, but have not been through the typesetting, pagination and proofreading processes, which may lead to differences between this version and the final one.

The final version of the manuscript will then appear on a regular issue of the journal.

Please cite this article as doi: 10.4081/jae.2025.1741

Submitted: 14 March 2024
Accepted: 13 February 2025

***Note**: The publisher is not responsible for the content or functionality of any supporting information supplied by the authors. Any queries should be directed to the corresponding author for the article.*

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article or claim that may be made by its manufacturer is not guaranteed or endorsed by the publisher.

Leveraging deep semantic segmentation for assisted weed detection

Francisco Garibaldi-Márquez,[1] Gerardo Flores,[2] Luis M. Valentín-Coronado[3,4]

[1]National Institute of Forestry, Agricultural and Livestock Research - Pabellón Experimental Field, Arteaga Pavilion, Aguascalientes, Mexico
[2]RAPTOR Lab, School of Engineering, College of Arts and Sciences, Texas A&M International University, Laredo, TX, USA
[3]Research Center in Optics, Leon, Guanajuato, Mexico
[4]National Council for Science and Technology (CONACyT), Mexico City, Mexico

Correspondence: Luis M. Valentín-Coronado, Centro de Investigaciones en Óptica A.C., Loma del bosque 115, Leon, 37150, Guanajuato, Mexico.E-mail: luismvc@cio.mx

Availability of Supporting Data: All data generated or analyzed during this study are included in this published article.

Competing Interests: The authors declare that they have no competing interests.

Financing: This research has not received any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Authors' Contributions: Built the dataset, wrote the program codes, analyzed the experimental data, and wrote the manuscript (F.G.M); Supervision, provided significant suggestions to the study, writing-review (G.F); Supervision, provided significant suggestions to the study, writing-review, and editing (L.M.V.C).

Abstract

In agriculture, it is crucial to identify and control weeds as these plant species pose a significant threat to the growth and development of crops by competing for vital resources such as nutrients, water, and light. A promising solution to this problem is adopting smart weed control systems (SWCS) that significantly reduce the use of harmful chemical products. Furthermore, SWCS leads to reduced production costs and a more sustainable and eco-friendly approach to farming. However, implementing SWCS in natural fields can be challenging, mainly due to difficulties in accurately localizing plants. To address this issue, a visual identification system can be employed to label plants from images using a process known as semantic segmentation. In this work, we have implemented, validated, and compared three deep learning approaches, including Mask Region-based Convolutional Neural Network (Mask R-CNN), Mask R-CNN enhanced with an Atrous Spatial Pyramid Pooling module (Mask R-CNN-ASPP), and a proposed model named Residual U-Net architecture, for the semantic pixel segmentation of high densities of both crops (*Zea mays*) and weeds (including narrow-leaf weeds and broad-leaf weeds). Data augmentation and transfer learning have also been implemented. The performance of the models was evaluated with the well-known metrics Precision, Recall, Dice similarity coefficient (DSC), and mean Intersection-Over-Union (mIoU). As a result of the analysis, the DSC and mIoU of Mask R-CNN-ASPP based models were up to 10.63% and 10.54% superior to that of the Mask R-CNN based models. Nonetheless, the proposed Residual U-Net architecture outperformed Mask R-CNN-ASPP based networks in all the metrics, reaching a DSC of 92.98% and mIoU of 87.12%. Thus, we have concluded that the proposed Residual U-Net-like architecture is the best alternative for the semantic segmentation task in images with high plant density. Our research addresses the challenge of weed identification and control in agriculture, helping farmers produce crops more efficiently while minimizing environmental impact.

Keywords: Deep-learning, semantic segmentation, corn and weeds identification, natural crop field conditions.

Introduction

Weed control is an indispensable practice in agriculture. One main reason for implementing this control is that weeds can compete with crops for essential resources like nutrients, sunlight, and water, which can significantly reduce crop yield (Picon *et al*., 2022). Moreover, weeds can act as carriers of pathogens, thereby augmenting the risk of disease infection to crop plants (Dentika *et al*., 2021). For instance, in the case of the corn crop with high densities of weeds left uncontrolled during the production cycle, its yield is reduced by up to 90% (Nedeljković *et al*., 2021). Thus, the identification and location of weeds in the crop field have emerged as a crucial research topic, enabling the implementation of a well-regarded approach to weed management, namely Site-Specific Weed Management (SSWM) (Montes de Oca *et al*., 2018; Montes de Oca and Flores, 2021a).

SSWM techniques involve the accurate detection and localization of weeds in the field to facilitate targeted control measures, such as the precise application of thermal beams or herbicide flow directly to weed-infested areas (Garibaldi-Márquez *et al*., 2022). This advanced technology has demonstrated remarkable results, leading to reductions of up to 82% in herbicide volume (Nikolić *et al*., 2021), decreased production costs (Monteiro and Santos, 2022), as well as a significant reduction in environmental pollution and herbicide residues in food (Montes de Oca and Flores, 2021b; Tang *et al*., 2016). Nevertheless, the first challenge for implementing an efficient SSWM technology is the discrimination of crops and weeds under natural conditions. Then, a computer vision system may perform weed and crop discrimination. An approach based on the pixel-wise semantic segmentation technique can be implemented in this context. Semantic segmentation aims to categorize each pixel in the image into a class, producing a segmentation map of the plant within the input image.

Traditional semantic segmentation algorithms, including normalized N-Cut (Shi and Malik, 2000), super-pixel (Ren and Malik, 2003), and k-means clustering (Arai and Barakbah, 2007), have been widely used in various classification tasks. However, when it comes to agricultural applications, these classical algorithms have exhibited certain limitations due to factors such as similarities in plant color, the complex shapes of leaves, and high plant densities (Zhang and Peng, 2022). To address these challenges, researchers in recent years have turned to more advanced segmentation techniques. Such techniques are commonly based on the use of deep learning (DL) architectures. These DL-based approaches have been overwhelmingly successful in several computer vision tasks including natural language processing, object recognition, and object segmentation, to name a few. Among the DL-based approaches, Convolutional Neural Networks (CNNs), which are the most popular deep learning architecture, have been utilized for plant classification (Garibaldi-Márquez *et al.*, 2022), disease detection (Jadhav *et al.*, 2021), nutrient deficiencies studies (Taha *et al.*, 2022), etc. The popularity of CNNs is because they can extract and learn multi-features from a set of input images (Picon *et al.*, 2022). Thus, special attention was paid to CNN models for semantic segmentation. For instance, in the work presented by Long *et al.* (2015), a Fully Convolutional Network (FCN) for segmentation has been proposed. In this work, the authors have shown that the multi-resolution layer combinations significantly improve the segmentation of objects present in the image while simultaneously simplifying and speeding up learning and inference. On the other hand, in the work reported by Ma *et al.* (2019), the authors segmented rice seedlings, weeds, and backgrounds utilizing the SegNet-FCN architecture, which was compared with the traditional FCN and U-Net networks. Similarly, in the work of Kolhar and Jayant (2021), the authors evaluated the residual U-Net, classic SegNet, and classic U-Net for segmentation of individual Arabidopsis and Tabacco plants, reaching a dice coefficient (DSC) of 97.09% from the residual U-Net. Regarding works carried out in corn crops, Dyrmann *et al.* (2016) classified soil, weeds, and corn plants based on FCN from RGB images, reporting an accuracy of over 94%. Recently, Picon *et al.* (2022) presented a modified PSPNet for segmenting corn plants, three narrow-leaf weed species, and three broad-leaf weed species. They reached a DSC of 25.32% when the plants were grouped into crop, narrow-leaf weeds, and broad-leaf weeds classes.

Even though numerous studies have explored the semantic segmentation of various crops, there needs to be more focus on segmenting one of the most crucial cereals, namely corn. Then, there is a clear need to propose and evaluate deep learning architectures specifically tailored for categorizing corn plants and distinguishing them from common weeds. Furthermore, to the best of our knowledge, the segmentation of corn plants in natural field conditions has received even less attention, an aspect of significant importance for understanding and accurately delineating corn plants within real-world agricultural environments, which is an essential requirement for developing site-specific weed and corn control systems.

In this work, a modified residual U-Net network specifically designed to achieve semantic segmentation of weeds and corn plants has been proposed. The performance of this proposed network has been compared with a Mask Region-based Convolutional Neural Network (Mask R-CNN) and a proposed improved version of the Mask R-CNN. The proposed Mask R-CNN (Mask R-CNN-ASPP) differs from the classic model by incorporating the *Atrous Spatial Pyramid Pooling* module.

The remainder of this work is organized as follows. Section 2 describes the problem statement and the experimental setup. Experiments conducted, under real conditions, that corroborate the main result and discussion of the findings are presented in Section 3 and Section 4 respectively. Finally, in Section 5 the conclusions of the work are presented.

Materials and Methods

Semantic segmentation is a computer vision technique that involves classifying each pixel in an image into a specific class or category, thereby dividing the image into meaningful segments. Unlike simpler forms of image segmentation, such as object detection, which identifies and locates objects in an image, semantic segmentation goes a step further by assigning a distinct label to every pixel, providing a detailed understanding of the image's content. This method is particularly valuable in various applications, including agriculture, where it plays a pivotal role in precision farming. Semantic segmentation helps identify and classify different elements within an agricultural scene, such as crops, soil, and weeds, enabling farmers to gain granular insights into their fields. Farmers often face challenges in accurately assessing crop health, identifying weeds, and optimizing resource allocation. Semantic segmentation can address these issues by enabling automated and precise delineation of crop boundaries, detection of plant diseases, and differentiation between crops and unwanted vegetation. In this context, the addressed problem can be summarized as follows:

Problem 1 *Given a ground-level set of images, denoted as* $X = \{x_i \in I^{m \times n \times 3} | 0 \leq i \leq N\}$*, where* $I^{m \times n \times 3}$ *refers to the set of* $m \times n$ *color images, acquired from a natural cornfield, the problem is to create a pixel-wise segmentation map where each pixel* $\left([x_i]_{r,c}, r = 1, \ldots, m; c = 1, \ldots, n\right)$ *is assigned to the class Corn plants (Crop), Narrow-leaf weeds (NLW), Broad-leaf weeds (BLW), or Soil, despite the presence of rocks, stubble, plant density, occlusions, shadows, and sunlight intensity. Thus, the proposed methodology aims to label each pixel of the input image by means of a deep-learning-based model* $(\mathcal{M})$ *with a specific class (Crop, NLW, BLW, Soil), i.e.,* $\mathcal{M}: [x]_{r,c} \mapsto y$*, where* $y \in \{Crop, NLW, BLW, Soil\}$*.*

To address Problem 1, we propose the overall process summarized in Figure 1, where an input image, which presents a high plant density, is pixel-wise segmented utilizing a deep-learning-based model (residual U-Net, Mask R-CNN, or Mask R-CNN-ASPP).

Dataset description and image pre-processing

We have collected extensive crop/weed images in natural corn fields to train and test the proposed system. All these images were pixel-level annotated. Most images were captured in a top-down view, and only a few were captured from side views. The image capture distance, h, between the plants and the camera was 0.4m to 1.5m, i.e., $h \in [0.4\text{m}, 1.5\text{m}]$. Most of the acquired images include different plant species (weeds) and several instances of the crop. The dataset for this study has been integrated by 10,200 images of sizes $4{,}608 \times 3{,}456$, $1{,}600 \times 720$, and $2{,}460 \times 1{,}080$ pixels. Our image dataset variability includes side views of plants and views of plants with zoom variation. Furthermore, several images were acquired with different light conditions, because these images were captured on sunny and cloudy days, in the morning, at noon, and in the afternoon. Additionally, our image dataset does not have a uniform background since elements like the soil appearance and straws from past crops were introduced. It is worth mentioning that the images were captured every five days. Figure 2 shows representative instances of the dataset. In the first row (Figure 2a), single-plant images are shown; in the second row (Figure 2b), multiple-plant images are provided, where leaves overlap, occlusion and soil appearance variability are observed. Finally, in the last row (Figure 2c), multiple small plants are depicted as a product of the maximum capture distance (h = 1.5m).

After carefully observing the acquired images, nine different plant species were found. These plants were grouped into three classes: i) Crop, ii) NLW, and iii) BLW. A manual labeling step using the tool VGG Image Annotator (Dutta and Zisserman, 2019) was conducted after grouping the plants in each of the 10,200 acquired images. This involved tracing carefully a polygon around the contour of most plants in the image, ensuring that soil pixels were consistently excluded. The built dataset is defined as $DS = \{(x_1, Y_i)\}_{i=1,\dots,N}$, where $x_i \in I^{m\times n\times 3}$ represents the *i-th* image and $Y_i = \{y_j\}_{j=1,\dots,M}$, in which $y_j \in \{Crop, NLW, BLW, Soil\}$, is the set of all labeled plants and soil in the *i-th* image. It is worth mentioning that, in general, $|Y_i| \neq |Y_k| \forall\ i \neq k$ ($|\cdot|$ refers to the cardinality of the set). Table 1 shows a summary of the plant species that belong to each class, the labels traced per plant species (LBLS), and the total number of labels for each class (LBLC). It is worth noticing that the class Soil is not reported in this table because it was indirectly annotated.

The proposed methodology aims to predict the elements "Y^*" presented in an input image "x" with a previously trained deep learning model.

Deep neural network architectures

The integration of deep neural networks and semantic segmentation has become a powerful technological advancement in the field of agriculture with diverse applications. Specifically, Convolutional Neural Networks (CNNs), a type of deep neural network, have proven to be adept at handling large amounts of agricultural data, enabling precise image analysis for various tasks (Dyrmann *et al.*, 2016; Ma *et al.*, 2019; Kolhar and Jayant, 2021). Through applying deep neural networks, semantic segmentation facilitates the accurate delineation of specific objects within agricultural images, such as crops, weeds, and soil.

Residual U-Net architecture

The proposed residual U-Net architecture consists of two key components: an encoder, also referred to as the backbone or contracting path, and a decoder or expansive path. The encoder performs convolutional operations to extract essential features. On the other hand, the decoder employs transposed 2D convolutional layers to upscale the feature blocks until they match the size of the original input image. In the proposed architecture, we utilize the ResNet50 and ResNet101 architectures (He *et al.*, 2016) to serve as the encoder part of our model. Figure 3 visually represents the proposed residual U-Net architecture.

From the input image, the encoder operations commence with a 7×7 padded convolution, followed by normalization and a Rectified Linear Unit (*ReLU*) activation function. These sequential operations yield an initial feature map with dimensions $256 \times 256 \times 64$. Subsequently, this feature map becomes the input for the "ResNet, B1" block, the output of which is then passed to the next "ResNet, B2" block. This process continues until the final output is obtained from the "ResNet, B4" block. Each ResNet block contributes to downsampling the feature maps, resulting in halved size and twice the number of channels compared to the previous stage, as depicted in Figure 3.

In the decoder section of our proposed network, we employ 3×3 transposed convolutions to facilitate the up-sampling of the feature maps at each step. This operation effectively doubles the size of the feature maps while reducing the number of channels by half. Consequently, the up-sampled feature maps are concatenated with the corresponding feature map obtained from the ResNet block at the same level in the encoder. Following the concatenation, two 3×3 padded convolutions and the *ReLU* activation function are applied. Lastly, at the final layer of the decoder, a 1×1 convolution is utilized to map each 64-dimensional feature vector to a four-channel output. This number of output channels aligns with the classes present in our dataset.

ResNet backbone details

The backbone of the residual U-Net architecture is constructed using ResNet50 and ResNet101 models. The selection of ResNet models was motivated by their demonstrated effectiveness in classifying plants in natural environments, as indicated by previous studies (Quan *et al.*, 2021; Peng *et al.*, 2022; Picon *et al.*, 2022; Zenkl *et al.*, 2022). The integration of ResNet50 and ResNet101 involves several components.

Figure 4a provides an overview of the entire ResNet blocks, including the arrangement of the residual blocks. As it can be appreciated, this structure begins with a 7×7 padded convolution layer with a stride of 2, followed by a 3×3 max pooling layer with the same stride. The subsequent structure comprises four consecutive main blocks, each consisting of residual blocks with unique properties. These main blocks are connected to a fully connected layer, which is then linked to the output layer responsible for generating the final predictions.

The presence of residual building blocks characterizes the ResNet module. Here, two types of residual blocks are utilized: the *identity block* (shown in Figure 4b) and the *convolutional block* (depicted in Figure 4c). The *identity block* is employed when the input feature map ($\mathfrak{m}$) and the output feature map of the block ($\varphi(\mathfrak{m})$) have the same dimensions.

As illustrated in Figure 4b, the *identity block* consists of three stacked convolutions ($1 \times 1, 3 \times 3, and\ 1 \times 1$), each followed by a normalization operation and a *ReLU* activation function. The resulting output is then element-wise added to the feature map ($\mathfrak{m}$) and fed into the residual block via a shortcut path. This addition yields the output $\mathcal{H}(\mathfrak{m})$, which represents the underlying mapping. Notably, the number of kernels used in the *identity block*, denoted as "C1" and "C2", varies depending on the specific main block (Block 1, Block 2, Block 3, or Block 4) within the ResNet architecture. For instance, in the first main block (Block 1), C1 = 64 and C2 = 256, while in the second main block (Block 2), C1 = 128 and C2 = 512, and so on. This variation allows the network to capture different levels of complexity and abstraction.

Compared to conventional CNNs that stack convolutional layers to approximate the input, the advantage of using residual blocks is that the network learns the residual map, expressed as $\varphi(\mathfrak{m}) = \mathcal{H}(\mathfrak{m}) - \mathfrak{m}$. This formulation helps to mitigate the vanishing gradient problem because if $\varphi(\mathfrak{m})$ tends to zero during back-propagation, the identity map m contributes to non-zero weights. Consequently, gradients can propagate to the initial layers of the network, allowing them to learn at a comparable rate to the final layers. This characteristic of residual blocks enables the training of deeper networks.

In scenarios where the input and output have different dimensions, the *convolutional block* is utilized. Unlike the identity block, the convolutional block incorporates a 1×1 convolutional layer in the shortcut path, in addition to the variation in the number of kernels. Specifically, for the convolutional block, the values of (C1, C2) are chosen from the set {(128, 512), (256, 1024), (512, 2048)}. It is important to note that the convolutional block is not present in the first main block (Block 1). Including the 1×1 convolutional layer in the shortcut path allows for adapting the dimensions of the feature maps to match the desired output size. This additional convolutional layer helps incorporate richer spatial information and adapt the network's capability to accommodate changes in spatial resolution throughout the network. However, in the first main block (Block 1), where the initial feature maps are obtained, the convolutional block is not required since the dimensions of the input and output feature maps are already compatible.

A notable distinction between ResNet50 and ResNet101 lies in the number of residual blocks within the main Block 3. Specifically, ResNet50 incorporates five residual blocks, while ResNet101 includes twenty-two residual blocks. Consequently, considering the shared 7×7 convolutional and 3×3 max pooling layers in both networks, the total number of layers in ResNet50 amounts to 50, whereas ResNet101 comprises 101 layers. The discrepancy in the number of residual blocks between the two architectures significantly impacts their depth and ability to capture intricate patterns and features within the input data. With a larger number of layers and residual blocks, ResNet101 possesses a more extensive and expressive network structure, facilitating the representation of increasingly complex relationships and enhancing its ability to learn hierarchical features. However, it is worth noting that the deeper architecture of ResNet101 may also introduce challenges, such as increased computational requirements and the potential risk of overfitting, especially in scenarios with limited training data. Consequently, the choice between ResNet50 and ResNet101 depends on the specific requirements of the task at hand, striking a balance between model complexity and computational efficiency.

Mask R-CNN architecture

Mask Region-based Convolutional Neural Network (Mask R-CNN) is a deep-learning architecture used for performing instance segmentation. This network can detect objects and "accurately" perform pixel-level instance segmentation on them. The illustration of Mask R-CNN is depicted in Figure 5.

The backbone of this network plays a critical role as it takes the input image and generates a feature map. Subsequently, a Region Proposal Network (RPN) analyzes this feature map to generate rectangular region proposals. However, it should be noted that these proposed regions derived from the feature map may be misaligned with respect to the input image. Thus, an ROI alignment process is employed to align these ROIs based on the input image. These components can be collectively summarized as the mapping function (f_θ) that transforms the input image into a fixed-size feature map. The head of the Mask R-CNN architecture comprises two parallel branches. The first branch is a fully connected layer, denoted as $f_\emptyset$, responsible for predicting and classifying bounding boxes for each ROI. The second branch is an FCN, denoted as f_γ, which predicts a binary mask for each class independently of the classification branch. The FCN consists of four consecutive 3×3 *conv* layers, followed by a 2×2 *deconv* layer with a stride of 2, and finally, a 1×1 *conv* layer. These hidden layers utilize the *ReLU* activation function. This configuration allows the segmenting of the objects in the image. To summarize the overall process, from each input image x_i, a feature map $\mathcal{F} = f_\theta(x_i)$ is computed. This feature map serves as the input for both the fully connected layers for feature extraction, $f_{\emptyset}(f_{\theta}(x_i))$, and the FCN $f_{\gamma}(f_{\theta}(x_i))$.

Mask R-CNN–ASPP architecture

The Mask R-CNN network, which employs convolutions and deconvolution operations in its segmentation branch, may have limitations due to the inability of convolutions to capture complete spatial context information from feature maps alone. Such information can be valuable for enhancing segmentation, particularly in scenarios with a high density of objects, as encountered in this study. To address this challenge and enhance the segmentation of corn and weed plants, we integrate the Atrous Spatial Pyramid Pooling (ASPP) module within the FCN branch of the Mask R-CNN architecture. The ASPP module leverages *atrous convolutions* (also known as dilated convolutions), which perform convolutions by incorporating pixels situated at a certain distance from the central pixel rather than using only adjacent pixels. This distance is defined by the dilatation rate (r). By employing *atrous convolutions*, the ASPP module enables the expansion of the filter's *field-of-view* (Chen *et al*., 2017). For this comparison, we introduce the ASPP module into the Mask R-CNN architecture, as illustrated in Figure 6.

As depicted in Figure 6, the ASPP module takes as input each fixed-sized feature map (ROI) computed by the ROIAlign block. The ASPP block applies three dilated convolutions and a pooling operation to each input ROI. The dilated convolutions have dilatation rates of one $(r = 1)$, three $(r = 3)$, and six $(r = 3)$respectively. Following the *atrous convolutions*, batch normalization and *ReLU* activation functions are applied. Additionally, the input ROI undergoes a 2×2 average pooling operation, followed by upsampling by a factor of 2 using bilinear interpolation. The outputs of these operations are concatenated and then convolved with a 1×1 kernel, followed by a *ReLU* activation function. This step results in a $14 \times 14 \times 256$ feature map. Subsequently, the remaining operations are consistent with the original FCN implementation of the Mask R-CNN architecture.

Metrics

A comprehensive set of metrics was employed to assess the overall performance of the networks, ensuring a thorough evaluation. The metrics utilized in this study included Precision, Recall, Dice Similarity Coefficient (DSC), Intersection over Union (IoU), and mean Intersection over Union (mIoU). The definitions of these metrics are summarized in Table 2, providing a clear understanding of how each metric contributes to the assessment process.

In addition to these metrics, the number of correct and incorrect predictions was recorded and analyzed using the widely recognized representation of the confusion matrix. This matrix offers valuable insights into the performance of the models by categorizing predictions into true positives, true negatives, false positives, and false negatives. By examining these values, we can understand the accuracy and efficacy of the models in differentiating between classes. We evaluate the networks' performance using various metrics and incorporating the confusion matrix representation. These evaluation measures provide valuable insights into the models' abilities to accurately identify and classify the target objects, facilitating informed decision-making and further improvements in object segmentation and classification.

Results

In this section, the experimental results for the pixel-wise semantic segmentation, as well as the overall performance of the proposed residual U-Net model, are presented. In addition, a comparison between the residual U-Net model and the two Mask R-CNN-based models is also shown.

Experimental setup

To evaluate the performance of the deep-learning models, a set of experiments, using our dataset, was carried out. Furthermore, to train the networks, the transfer learning method has been implemented. Transfer learning refers to a method where a pre-trained model, developed for a similar task, is reused as the starting point for a new task, thus allowing rapid and improved performance. For this work, the pre-trained models of the ResNet50 and ResNet101 networks in the well-known ImageNet dataset (Krizhevsky *et al.*, 2012) were used. It is worth mentioning that a desktop computer with an Intel Core i7 processor, NVIDIA GPU GeForce GTX 1080Ti with 6 GB of VRAM, and 64 GB of RAM memory was used to re-train the models. The implementation was carried out in Python 3.8 and Keras framework with Tensorflow 2.4.0 as a backend.

Training of the deep neural networks

For training the models, we split the dataset into 70% for training, 20% for validation, and 10% for testing. It is worth mentioning that these images were randomly selected with uniform probability. Moreover, to ensure equal representation of instances per plant class, the dataset was balanced, resulting in 22,622 instances per class. Furthermore, a batch size of one and 200 epochs was established to train all models.

Residual U-Net training

For training this model, each input image, $x \in I^{n \times m \times 3}$, was resized to a dimension of $512 \times 512 \times 3$ $\left(\mathcal{S}: I^{n \times m \times 3} \longrightarrow I_{\mathcal{S}}^{512 \times 512 \times 3}\right)$. Subsequently, the image was mapped using the function $\mathcal{L} : I_{\mathcal{S}}^{512 \times 512 \times 3} \longrightarrow [0,1]^{512 \times 512 \times 3} \cap \mathbb{R}^{512 \times 512 \times 3}$, normalizing each pixel value to the range $[0,1] \cap \mathbb{R}$.

In addition, the Stochastic Gradient Descent (SGD) optimizer with a learning rate of 0.0001 was configured. The dice loss function was implemented to calculate the error between the ground truth image and the predicted mask image. On the other hand, the focal loss function was used to compensate for the complicated finding of the NLW class pixels, since it usually occupies a big area but a low number of pixels in the image, due to the phenological appearance of the plant species.

The computation of dice loss is as follows,

$$L_{Dice} = 1 - \frac{2yy^{*} + 1}{y + y^{*} + 1} \tag{1}$$

where y refers to the ground truth label and y^{*} is the predicted value from the model.

Respecting the categorical focal loss, it is computed as follows,

$$L_{Focal} = -\alpha_t (1 - p_t)^{\emptyset} \log(p_t) \tag{2}$$

where $\alpha_t \in [0,1]$ is a vector of class weights which is computed as the inverse class frequency from the dataset labels, p_t is a matrix of probabilities that each class has to be ground truth, and ϕ is the degree of modulating the pixels that are easy to classify (usually $\phi = 2$).

Mask R-CNN and Mask R-CNN-ASPP training

Mask R-CNN and Mask R-CNN-ASPP are specialized in instance segmentation of objects. Each segmented object in our models is associated with its corresponding class (Crop, NLW, NLB) to ensure a clear image interpretation. In other words, if an image contains “n” objects, each object is classified as either crop, narrow-leaf weeds, or broad-leaf weeds.

These architectures were configured to support input images $x \in I^{n \times m \times 3}$ with maximum dimensions of $1024 \times 1024 \times 3$. As some images in the dataset contained around 250 labels, the Region Proposal Network (RPN) was trained with 500 anchor boxes and Regions of Interest (ROIs) per image. During training, the weight decay and the learning rate were 0.0001, with the optimizer chosen as SGD.

To compute the suitable network parameters, the loss function proposed by He *et al.* (2017), has been used. This loss function is defined as follows,

$$L = L_{cls} + L_{box} + L_{mask} \tag{3}$$

where L represents the total loss function of the model, L_{cls} is the classification loss, L_{box} is the bounding box regression loss, and L_{mask} is the average binary cross-entropy loss. Particularly, the classification loss (L_{cls}) is computed according to,

$$L_{cls} = \frac{1}{N_{cls}} \sum_{i} -log[p_i p_i^{*} + (1 - p_i^{*})(1 - p_i)] \tag{4}$$

where N_{cls} are the number of categories, p_i is the probability that the *i-th* ROI is predicted to be the target. Here, when the predicted ROI is foreground $p_i^{*} = 1$, otherwise $p_i^{*} = 0$.

On the other hand, the bounding box regression loss (L_{box}) is computed by the following expression,

$$L_{box} = \frac{1}{N_{box}} \sum_{i} p_i^* R(t_i, t_i^*) \qquad (5)$$

where N_{box} is the number of pixels in the feature map, $R(\cdot)$ is a smooth function, t_i represents the four parameterized coordinate vectors of the predicted ROIs, and t_i^* indicates the coordinate vector of the real label.

Finally, the mask loss (L_{mask}) calculation is given by,

$$L_{mask} = -\frac{1}{N} \sum_{i} \left[y_i^* \log\left(p(y_i)\right) - (1 - y_i^*) \log\left(1 - p(y_i)\right) \right] \qquad (6)$$

where N represents the number of pixels, p_i^* is the predicted *k-th* class in that pixel's location, and $p(y_i)$ is the probability of the y_i predicted category.

Behavior of loss functions and mIoU during training

The behavior of the loss functions in the training stage for the deep learning models can be seen in Figure 7. The green and black curves show the training behavior of the proposed residual U-Net network with ResNet50 and ResNet101 backbones, respectively. The red and blue curves correspond to Mask R-CNN-ResNet50 and Mask R-CNN-ResNet101, respectively. Similarly, the magenta and cyan curves represent Mask R-CNN-ASPP-ResNet50 and Mask R-CNN-ASPP-ResNet101, respectively.

From Figure 7, it may be observed that the Mask R-CNN-based architectures' training and validation error curves oscillate a bit during the entire training process. In contrast, the error curves of the residual U-Net model present a monotonically decreasing behavior. Note that the oscillation error of the Mask R-CNN-based models may be attributed to the dependency on the mask, class, and box loss functions used during training, as these networks depend on the "correctly" detecting ROIs, which is carried out by the RPN block.

According to Figure 7, it can also be observed that the overall error of the Mask R-CNN-ASPP architectures had a lower magnitude during all the training steps than the error of the original Mask R-CNN architectures. In particular, Mask R-CNN-ASPP-ResNet50 had better loss behavior.

On the other hand, the mIoU metric allows a straight evaluation of a segmentation model's performance because it indicates the overlap of the prediction mask over the ground truth. Therefore, the value of this metric over the validation data at each epoch was registered. Figure 8 shows the behavior of the mIoU metric. It can be appreciated that the highest value was given by residual U-Net-ResNet101 during the training process, followed by residual U-Net-ResNet50. The mIoU of the Mask R-CNN-ASPP networks in all epochs was superior to that of the original Mask R-CNN architecture.

Performance of deep learning models over the classes

To evaluate the performance of each model, 10% of the data from the entire dataset was used. Note that none of these data was used during training. The classification results achieved by each model are presented in Table 3.

From Table 3, it can be appreciated that the Mask R-CNN-ASPP models have shown better Precision than the original Mask R-CNN models. Among classes, Mask R-CNN- ASPP-ResNet101 depicted higher Precision values than Mask R-CNN-ASPP-ResNet50, except for class soil, where Mask R-CNN-ASPP-ResNet50 is slightly superior. In this way, the Precision shown by Mask R-CNN-ASPP-ResNet101 for the classes Crop, NLW, and BLW was 5.75%, 19.88%, and 20.39% higher, with respect to values obtained by the Mask R-CNN-ResNet50 model. However, the Precision values of the Mask R-CNN- ASPP models were also surpassed by those obtained by the residual U-Net model. The best residual U-Net model was the one whose backbone was ResNet101, whose Precision was 7.13%, 19.34%, and 1.94% higher than that of Mask-R-CNN-ASPP-ResNet101 for the classes Crop, NLW, and BLW, accordingly.

Regarding the Recall metric among the plant classes, the Mask R-CNN-ASPP-ResNet50 model has performed better than the Mask R-CNN-ASPP-ResNet101 model, except for the class NLW. In the case of the Mask R-CNN models, this metric was better yielded by Mask R-CNN-ResNet50. Nevertheless, the Recall value depicted by Mask R-CNN-ASPP-ResNet50 for the classes Crop, NLW, and BLW was 22.18%, 19.28%, and 5.83% superior, respectively, to that shown by Mask R-CNN-ResNet50. However, as observed in Table 3, the Recall exhibited by residual U-Net-ResNet101 surpassed that of the Mask R-CNN-ASPP-ResNet50 model in the three plant classes. Then, residual U-Net-ResNet101 exhibited 23.28%, 38.03%, and 14.08% higher Recall rates than Mask R-CNN-ASPP-ResNet50 for the Crop, NLW, and BLW classes respectively.

Concerning the behavior of the DSC metric, the two Mask R-CNN-ASPP models have obtained higher DSC values than the Mask R-CNN models. However, the performance of the two Mask-R-CNN-ASPP models was surpassed by the residual U-Net models. The DSC of Mask R-CNN-ASPP-ResNet50, which was better than Mask-R-CNN-ASPP- ResNet101, was 20.88%, 19.75%, 2.64% superior for the classes, Crop, NLW, and BLW, accordingly, than that obtained by Mask R-CNN-ResNet50, which was better than Mask R-CNN-ResNet101. Nevertheless, residual U-Net-ResNet101, which was better than residual U-Net-ResNet50, manifested a superior DSC in 17.33%, 31.15%, and 17.78% for the classes Crop, NLW, and BLW, respectively, than that of Mask R-CNN-ASPP-ResNet50.

Finally, Mask R-CNN-ResNet50 obtained better IoU than Mask R-CNN-ResNet101 for all the plant classes. However, both overcame Mask R-CNN-ResNet50, the Mask R-CNN-ASPP-ResNet50, and the Mask R-CNN-ASPP-ResNet101 models. The IoU metric, obtained by Mask R-CNN-ASPP-ResNet50, demonstrated a significant increase in performance compared to Mask R-CNN-ResNet50. Specifically, we observed an increase of 22.88%, 17.04%, and 3.63% for the Crop, NLW, and BLW classes, respectively. Nevertheless, similar to the other metrics, the IoU of the residual U-Net models was better than that of the Mask R-CNN-ASPP-based models for all the plant classes. In this way, the IoU obtained by residual U-Net-ResNet101 was 25.73%, 39.16%, and 17.93% better for the classes Crop, NLW, and BLW, respectively, than that obtained by Mask R-CNN-ASPP-ResNet50.

The suggested Mask R-CNN-ASPP models generally perform better than the Mask R-CNN models. Nevertheless, the performance of Mask R-CNN-ASPP-based models was overcome by the proposed residual U-Net models.

Semantic segmentation comparison

The performance of the trained model can be summarized by the confusion matrices in Figure 9. As it can be appreciated, all the models have generally classified the soil with a classification rate of over 97%. This behavior may be attributed to their predominance in the images.

Regarding the pixel classification of Crop, NLW, and BLW, pixels belonging to the class BLW were the best classified by all the models, and the worst was the NLW class's pixels. The high classification rate of the pixels belonging to the class BLW is attributed to the phenological appearance of the plant species that integrate this group. Note that this class differs morphologically from the plant species that integrate Crop and NLW. In contrast, the low-rate classification of the pixels belonging to the class NLW is also attributed to the phenological appearance of the plant that integrates the group because they are narrow and occupy a low area in the images, often classified as soil pixels.

It is also observed from Figure 9, that Mask R-CNN-ResNet50 classified the pixels of the three plant classes better than Mask R-CNN-ResNet101. On the other hand, the Mask R-CNN-ASPP-ResNet50 model has classified the pixels of the classes Crop and BLW better than the Mask R-CNN-ASPP-ResNet101 but not the pixels of the class NLW.

In the case of the proposed residual U-Net networks, when the ResNet50 is used as the backbone, the model correctly classified the pixels of the class BLW. Nevertheless, in the case of ResNet101, the model better classified the pixels of the classes Crop and NLW.

It is appreciated that Mask R-CNN-ASPP-ResNet50 was 5.83% higher at recognizing BLW pixels than Mask R-CNN-ResNet50, contrasting the better model from each of the three groups on classifying the pixels. However, residual U-Net-ResNet50 was 14.54% better at classifying the pixels of BLW than Mask R-CNN-ASPP-ResNet50. Mask R-CNN-ASPP-ResNet50 was 22.18% superior at recognizing Crop pixels as such than Mask R-CNN-ResNet50. However, residual U-Net-ResNet101 surpassed in 23.28% the recognition of the pixels of this class than Mask R-CNN-ASPP-ResNet50. Finally, residual U-Net-ResNet101 classified 56.31% and 37.29% better the class NLW, compared to Mask R-CNN-ResNet50 and Mask R-CNN-ASPP-ResNet101, accordingly.

In summary, from Figure 9, it is observed that all the models, some in a high percentage, misclassified as soil the pixels of the plant classes. Also, all the models confused the pixels of the class Crop with that of the class NLW, and vice versa. This behavior may also be imputed to the phenological appearance of the plants; since they are monocotyledonous plants, as consequence they may share some features.

Discussion

The average performance of each model, regarding the evaluation metrics, is shown in Figure 10. From all the evaluated metrics (Precision, Recall, DSC, and mIoU), the proposed residual U-Net networks outreached Mask R-CNN-ASPP-based and Mask R-CNN-based networks on semantic segmentation of the classes Crop, NLW, BLW, and Soil of our dataset. Nonetheless, the two Mask R-CNN-ASPP-based models overcome the performance of the Mask R-CNN-based models. However, residual U-Net-ResNet101 achieved the highest values of all the metrics.

Regarding the performance of Mask R-CNN-based models, Mask R-CNN-ResNet50 performs better than Mask R-CNN-ResNet101, which also turned out to be the one with the lowest performance. Lastly, Mask R-CNN-ASPP-ResNet50, in general, segmented our dataset better than Mask R-CNN-ASPP-ResNet101, as three out of the four metrics indicate.

The residual U-Net-ResNet101 model achieved the highest Precision value (93.79%). This was significantly greater than the Precision values of other models, including residual U-Net-ResNet50, Mask R-CNN-ResNet50, Mask R-CNN-ResNet101, Mask R-CNN- ASPP-ResNet50, and Mask R-CNN-ASPP-ResNet101, by 2.52%, 19.21%, 23.63%, 10.4%, and 8.39% respectively. The Precision value is an important metric that indicates the ability of the models to accurately classify each pixel of the images into the corresponding Crop, NLW, BLW, and Soil ground truth.

The best network model for Recall's case was the residual U-Net-ResNet101, achieving a remarkable value of 92.23%. Notably, the Recall of this network was 19.18% and 30.58% higher than that obtained by R-CNN-ASPP-ResNet50 and R-CNN-ResNet50, respectively, which were the best-performing networks in their respective categories. Concerning the DSC metric, R-CNN-ASPP-ResNet50 outperformed R-CNN-ResNet50 by 10.63%. Nevertheless, the residual U-Net-ResNet101 network, which performed optimally overall, displayed a DSC value 15.37% superior to that of R-CNN-ASPP-ResNet50. Lastly, the mIoU of residual U-Net-ResNet101 was 32.34% and 31.8% better than that of Mask R- CNN-ResNet50 and R-CNN-ASPP-ResNet50, respectively, which were the most effective networks in their categories.

Visualization of segmented classes

The visualization of the segmentation output of any model reinforces the comprehension of the numerical metrics. Therefore, a qualitative comparison from the segmentation output of the Mask

R-CNN-ResNet50, Mask R-CNN-ASPP-ResNet50, and residual U-Net-ResNet101, which were the network architectures with the best results, is presented in Figure 11. In the first row, the input image is shown (Figure 11a). The second row (Figure 11b) shows the ground truth in which the colors green, red, and blue represent the Crop, NLW, and BLW classes, respectively. Subsequently, the third row (Figure 11c) presents the segmentation output of the Mask R-CNN-ResNet50 model, whereas the fourth row (Figure 11d) shows the segmentation carried out by the Mask R-CNN-ASPP-ResNet50 model, and finally, in the last row (Figure 11d) the segmentation output of residual U-Net-ResNet101 model is shown.

As can be observed, the three models have segmented each class correctly when the plants are "separated" from each other and when the objects in the image are big enough. These conditions commonly occur when the image has been captured at a short distance, as is depicted in the first column of Figure 11. Regarding the segmentation performed by the Mask R-CNN-ResNet50 and Mask R-CNN-ASPP-ResNet50, it can be noticed that both models tend to fail when there are more than two plant classes, when the plants are close to each other and when plants appear small in the images, as may be observed in the second, third and fourth column of Figure 11c and Figure 11d. Also, these images give an insight into how the Mask R-CNN-ResNet50 and Mask R-CNN-ASPP-ResNet50 models commonly confuse pixels belonging to the class NLW with the Soil class. Nonetheless, in the image of the fourth column of Mask R-CNN-ASPP-ResNet50 model, the class NLW has been correctly segmented, attributed to the ASPP module implemented in its segmentation branch. The compilation of images presented here demonstrates that the residual U-Net-ResNet101 model yields superior segmentation outcomes, as evidenced by the near-perfect fit of its output masks with the ground truth data acquired in real-world field settings.

Comparison with state-of-the-art methods

Even though detecting common weeds that grow in corn fields is challenging, scarce works have been reported in natural conditions at high-density plants and addressed by semantic segmentation approaches. In the work of Fawakherji *et al.* (2020), the original U-Net architecture (Ronneberger *et al.*, 2015) and U-Net with VGG16 network (Simonyan and Zisserman, 2015) as the backbone (U-Net-VGG16) were evaluated. They reported a mIoU of 62% and 64%, for U-Net and U-Net-VGG16 respectively, when these models were trained with a Sunflower dataset and tested over the combined datasets Carrots and SugarBeets. The classes were crop, weed, and soil. Then, they evaluated the U-Net-VGG16 over the individual datasets SugarBeets, Stuttgart, Carrots, and Sunflower, reporting 71%, 45%, 35%, and 39% of mIoU, respectively. Although this work has not been done in corn crops, the databases were generated under natural conditions. In this way, the mIoU of our best model, residual U-Net–ResNet101, is 16.12% higher than the U-Net-VGG16 model reported in Fawakherji *et al.* (2020).

Other related works on semantic segmentation of crop plants and weeds are presented in Table 4. Even though the crops and trained architectures differ from ours, they also share the complexity of training the deep learning models using datasets acquired in natural environments. Therefore, the parameters dataset size, number of plant species in the dataset, DSC, and the mIoU have been highlighted to contrast them with our work. In this case, our work stands out from the others because a dataset with 10,200 images and nine plant species has been used. Furthermore, our proposed model has achieved superior performance compared to the most related state-of-the-art works, as demonstrated by the DSC and mIoU metrics. Table 4 shows that the datasets of the related works contain fewer images than our dataset. Increasing the number of images and plant species also increases the number of features the models need to learn, making the task more challenging. Our mIoU was 25.32% higher than that reported by Ma *et al.* (2019), despite their dataset being smaller. Khan *et al.* (2020) also used a reduced dataset with two plant species; their reported DSC and mIoU were 12.9% and 16.07% lower than those obtained by our best model. Among the works listed in Table 4, Zenk *et al.* (2022) reported the highest metric values; however, they only segmented wheat crops. Additionally, although Kamath *et al.* (2022) and Picon *et al.* (2022) increased their datasets, however, these are on average 80% smaller than ours. The mIoU obtained by Kamath *et al.* (2022) was 24.69% lower than ours. Conversely, the DSC of Picon *et al.* (2022), whose dataset contains seven classes, was 67.66% lower than that reached by our model.

Conclusions

This work proposes a residual U-Net network for semantic segmentation of crop and weed plants under real natural field conditions. The implemented residual U-Net network was built using a ResNet-based block in the encoding stage (backbone).

The experimental dataset used is made up of 10,200 images containing 59,681 labels, from which 18,423 are Crop, 18,636 are BLW, and 22,622 are NLW. These images have been captured under non-controlled conditions and have also been manually annotated.

For comparison purposes, two different deep learning models with corresponding variations have been used to analyze the experimental dataset, including Mask R-CNN and an enhanced Mask R-CNN. The enhanced Mask R-CNN (denoted as Mask R-CNN-ASPP) uses an Atrous Spatial Pyramid Pooling (ASPP) module implemented over the segmentation branch of the Mask R-CNN model. Mainly, ResNet50 and ResNet101 were the used architectures. Hence, six different networks have been implemented: the two proposed models, residual U-Net-ResNet50 and residual U-Net-ResNet101, together with four models used for comparisons: Mask R-CNN-ResNet50, Mask R-CNN-ResNet101, Mask R-CNN-ASPP-ResNet50, and Mask R-CNN-ASPP-ResNet101.

Experimental results have shown that the performance of the two Mask R-CNN-ASPP models overcomes the performance of the Mask R-CNN models. Nonetheless, the performance of the two

Mask R-CNN-ASPP models has also been outreached by the performance of the two proposed residual U-Net models. In particular, residual U-Net-ResNet101 was the best network, achieving a performance of 92.98% and 87.12% in terms of the metrics Dice coefficient (DSC) and mean intersection over Union (mIoU), respectively. These results are 15.57% and 21.8% better than the reached by the Mask R-CNN-ASPP-ResNet50 network, which was the second-best Mask R-CNN-ASPP-based model.

Regarding the plant classes, the experimental results have consistently demonstrated that the models achieved the highest accuracy in classifying pixels belonging to the broad-leaf weeds (BLW) class. In particular, the pixels representing NLW were often misclassified as Soil pixels, indicating a higher degree of confusion between these two classes compared to the other classes.

In future work, an increase in the number of elements of the dataset, with annotated labels, will be made so that it is possible to have a balanced dataset (same amount of data per class) and thus avoid using the data augmentation technique.

References

Arai, K., Barakbah, A.R. 2007. Hierarchical k-means: an algorithm for centroids initialization for k-means. Rep. Fac. Sci. Eng. 36:22-31.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L. 2017. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE T. Pattern Anal. 40:834-848.

Dentika, P., Ozier-Lafontaine, H., Penet, L. 2021. Weeds as pathogens hosts and disease risk for crops in the wake of a reduced use of herbicides: Evidence from yam (Dioscorea alata) fields and Colletotrichum pathogens in the tropics. J. Fungi 7:283.

Dutta, A., Zisserman, A. 2019. The VIA annotation software for images, audio and video. MM ’19, Proc. 27th ACM Int. Conf. on Multimedia, New York, pp. 2276-2279.

Dyrmann, M., Mortensen, A.K., Midtiby, H.S., Jørgensen, R.N. 2016. Pixel-wise classification of weeds and crops in images by using a fully convolutional neural network. In: CIGR-AgEng conference, Aarhus. Available from: https://conferences.au.dk/uploads/tx_powermail/cigr2016paper_semanticsegmentation.pdf

Fawakherji, M., Youssef, A., Bloisi, D.D., Pretto, A., Nardi, D. 2020. Crop and weed classification using pixel-wise segmentation on ground and aerial images. Int. J. Robot. Comput. 2:39-57.

Garibaldi-Márquez, F., Flores, G., Mercado-Ravell, D.A., Ramírez-Pedraza, A., Valentín-Coronado, L.M. 2022. Weed classification from natural corn field-multi-plant images based on shallow and deep learning. Sensors (Basel) 22:3021.

He, K., Gkioxari, G., Dollár, P., Girshick, R. 2017. Mask R-CNN. Proc. 2017 IEEE Int. Conf. on Computer Vision (ICCV), Venice; pp. 2980-2988.

He, K., Zhang, X., Ren, S., Sun, J. 2016. Deep residual learning for image recognition. Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Las Vegas; pp. 770-778.

Jadhav, S.B., Udupi, V.R., Patil, S. 2021. Identification of plant diseases using convolutional neural networks. Int. J. Inf. Tecnol. 13:2461-2470.

Kamath, R., Balachandra, M., Vardhan, A., Maheshwari, U. 2022. Classification of paddy crop and weeds using semantic segmentation. Cogent Engin.9:2018791.

Khan, A., Talha, I., Umraiz, M., Ibna Mannan, Z., Kim, H. 2020. Ced-net: Crops and weeds segmentation for smart farming using a small cascaded encoder-decoder architecture. Electronics (Basel) 9:1602.

Kolhar, S., Jayant, J. 2021. Convolutional neural network based encoder-decoder architectures for semantic segmentation of plants. Ecol Inform, 64:101373.

Krizhevsky, A., Sutskever, I., Hinton, G. 2012. ImageNet classification with deep convolutional neural networks. In: F. Pereira, C.J.C. Burges, L. Bottou and K. Weinberger (eds.), Adv. in Neural Inform. Process. Syst. 25. Curran Associates, Inc. Lake Tahoe.

Long, J., Shelhamer, E., Darrell, T. 2015. Fully convolutional networks for semantic segmentation. Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston; pp. 3431-3440.

Ma, X., Deng, X., Qi, L., Jiang, Y., Li, H., Wang, Y., Xing, X. 2019. Fully convolutional network for rice seedling and weed image segmentation at the seedling stage in paddy fields. PLoS One 14:e0215676.

Monteiro, A,. Santos, S. 2022. Sustainable approach to weed management: The role of precision weed management. Agronomy (Basel) 12:118.

Montes de Oca, A., Arreola, L., Flores, A., Sánchez, J., Flores, G. 2018. Low-cost multispectral imaging system for crop monitoring. Proc. Int. Conf. on Unmanned Aircraft Systems (ICUAS), Dallas; pp. 443-451.

Montes de Oca, A., Flores, G. 2021a. The AgriQ: A low-cost unmanned aerial system for precision agriculture. Expert Syst. Appl. 182:115–163.

Montes de Oca, A., Flores, G. 2021b. A UAS equipped with a thermal imaging system with temperature calibration for crop water stress index computation. Proc. Int. Conf. on Unmanned Aircraft Systems (ICUAS), Athens; pp. 714-720.

Nedeljković, D., Knežević, S., Božić, D., Vrbnićanin, S. 2021. Critical time for weed removal in corn as influenced by planting pattern and pre-herbicides. Agriculture (Basel) 11:587.

Nikolić, N., Rizzo, D., Marraccini, E., Ayerdi Gotor, A., Mattivi, P., Saulet, P., Persichetti, A., Masin, R. 2021. Site- and time-specific early weed control is able to reduce herbicide use in maize- a case study. Ital. J. Agron. 16:1780.

Peng, H., Li, Z., Zhou, Z., Shao, Y. 2022. Weed detection in paddy field using an improved

RetinaNet network. Comput. Electron. Agr. 199:107179.

Picon, A., San-Emeterio, M., Bereciartua-Perez, A., Klukas, C., Eggers, T., ad Navarra- Mestre, R. 2022. Deep learning-based segmentation of multiple species of weeds and corn crop using synthetic and real image datasets. Comput. Electron. Agr. 194:106719.

Quan, L., Wu, B., Mao, S., Yang, C., Li, H. 2021. An instance segmentation-based method to obtain the leaf age and plant centre of weeds in complex field environments. Sensors (basel) 21:3389.

Ren, X. Malik, J. 2003. Learning a classification model for segmentation. Proc. 9th IEEE Int. Conf. on Computer Vision, Nice; pp. 10-17.

Ronneberger, O., Fischer, P., Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. Wells, and A. Frangi (eds.), Medical image computing and computer-assisted intervention Vol. 9351. Lecture Notes in Computer Science. Cham, Springer. pp 234-241.

Shi, J., Malik, J. 2000. Normalized cuts and image segmentation. IEEE T. Pattern Anal. 22:888-905.

Simonyan, K., Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. Proc. 3rd Int. Conf. on Learning Representations, San Diego.

Taha, M.F., Abdalla, A., ElMasry, G., Gouda, M., Zhou, L., Zhao, N., et al. 2022. Using deep convolutional neural network for image-based diagnosis of nutrient deficiencies in plants grown in aquaponics. Chemosensors (Basel) 10:45.

Tang, J.L., Chen, X.Q., Miao, R.H., Wang, D. 2016. Weed detection using image precision under different illumination for site-specific areas spraying. Comput. Electron. Agr. 122:103-111.

Zenkl, R., Timofte, R., Kirchgessner, N., Roth, L., Hund, A., Van Gool, L., et al. 2022. Outdoor plant segmentation with deep learning for high-throughput field phenotyping on a diverse wheat dataset. Front. Plant Sci. 12:774068.

Zhang, H., Peng, Q. 2022. Pso and k-means-based semantic segmentation toward agricultural products. Future Gener. Comp. Sy. 126:82–87.

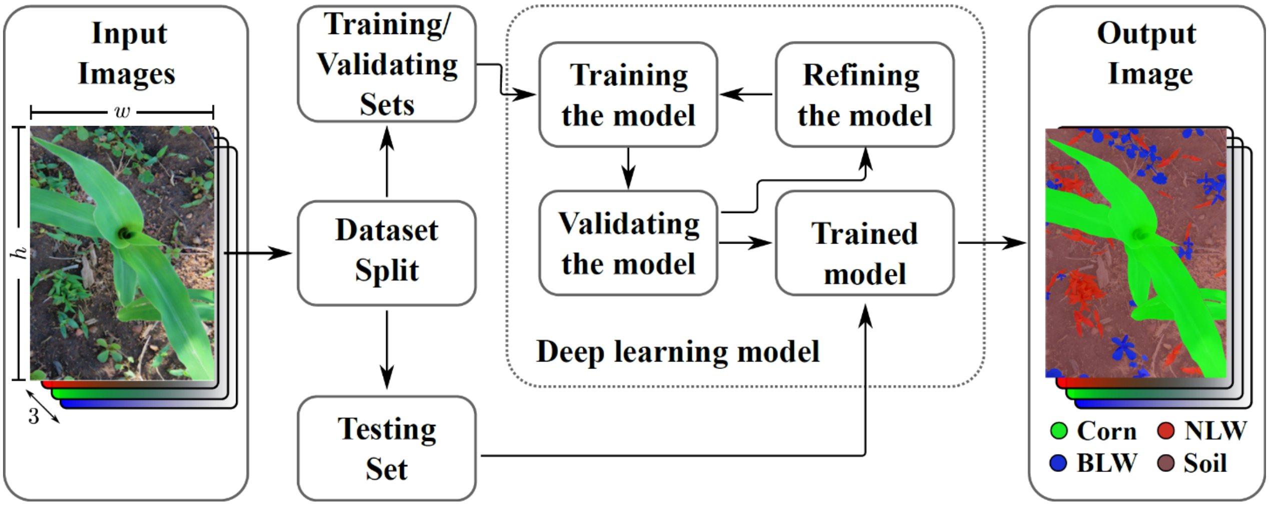
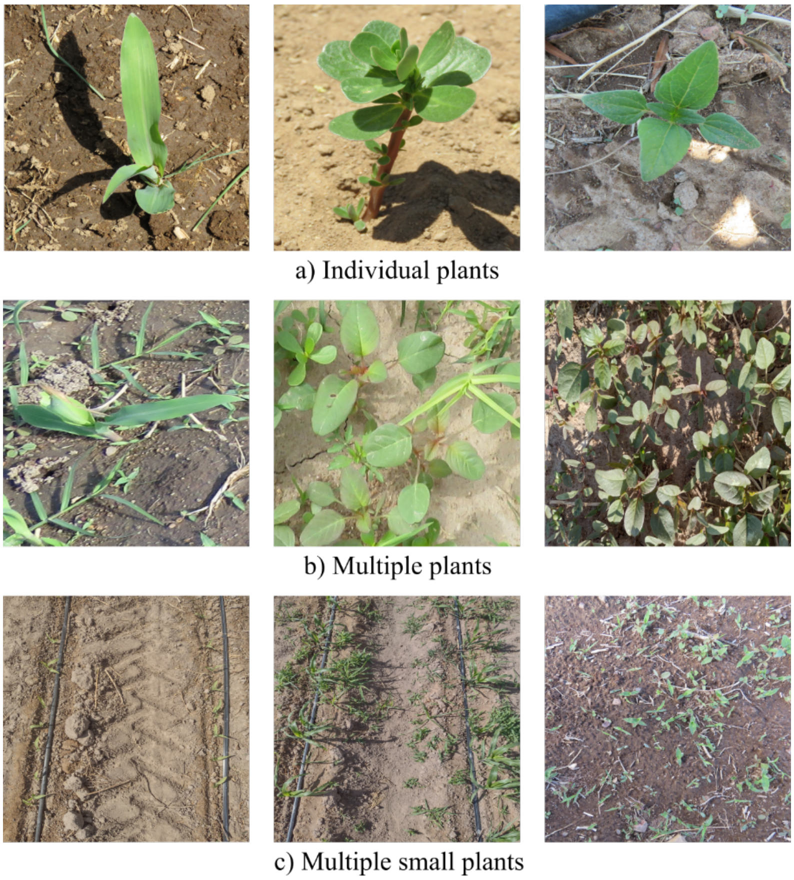


Figure 1. Methodology overview for crop/weed semantic segmentation in natural corn fields. The ground-level input image is shown on the left-hand side. In the center, the training/testing deep learning model is presented. On the right-hand side, the semantic segmented image is shown. This output image shows in green the segmented crop plant, in red the narrow-leaf weeds (NLW), and in blue the broad-leaf weeds (BLW).



a) Individual plants

b) Multiple plants

c) Multiple small plants

Figure 2. Sample images of our dataset. The first row shows individual plant images, while the second row features images of multiple plants, exhibiting scenarios of overlapping leaves, occlusion, and soil appearance variability. Finally, the last row depicts images of multiple small plants, likely captured from the maximum possible distance.

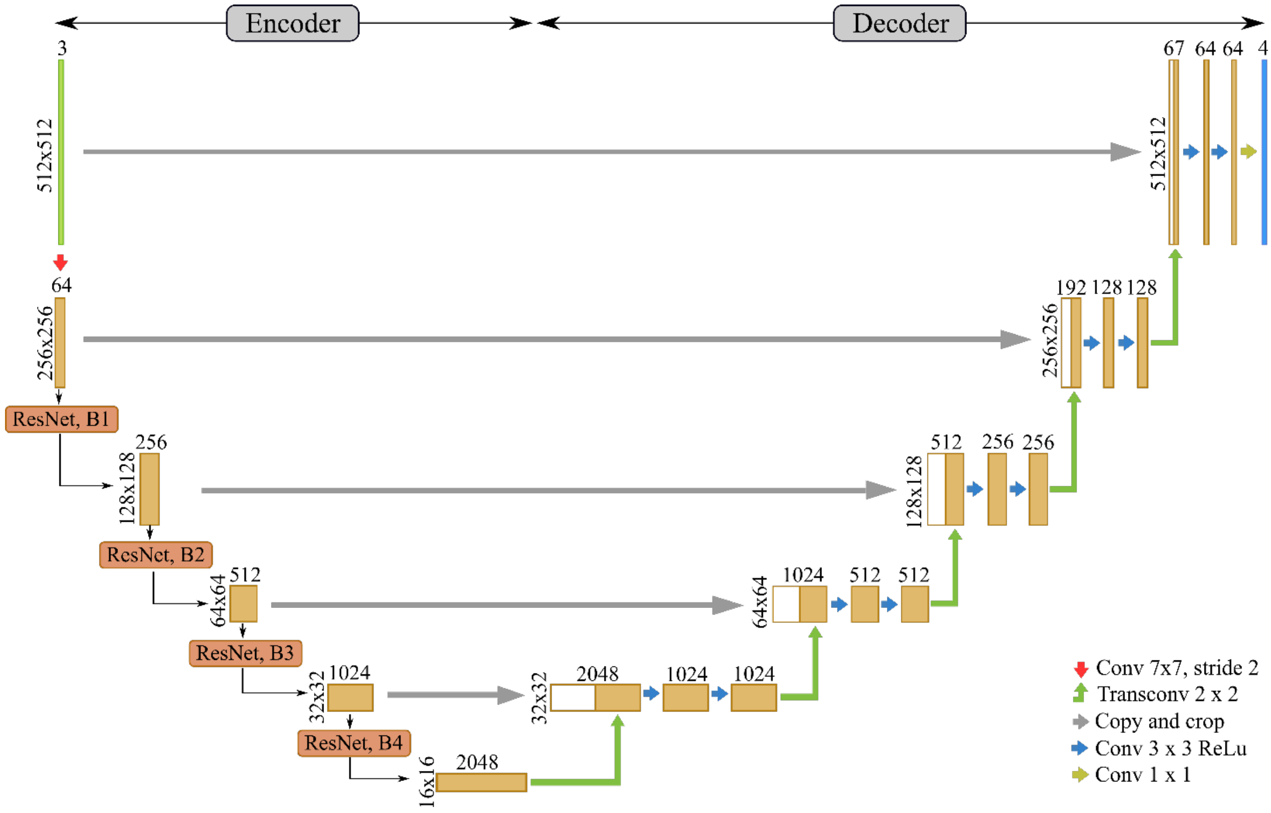
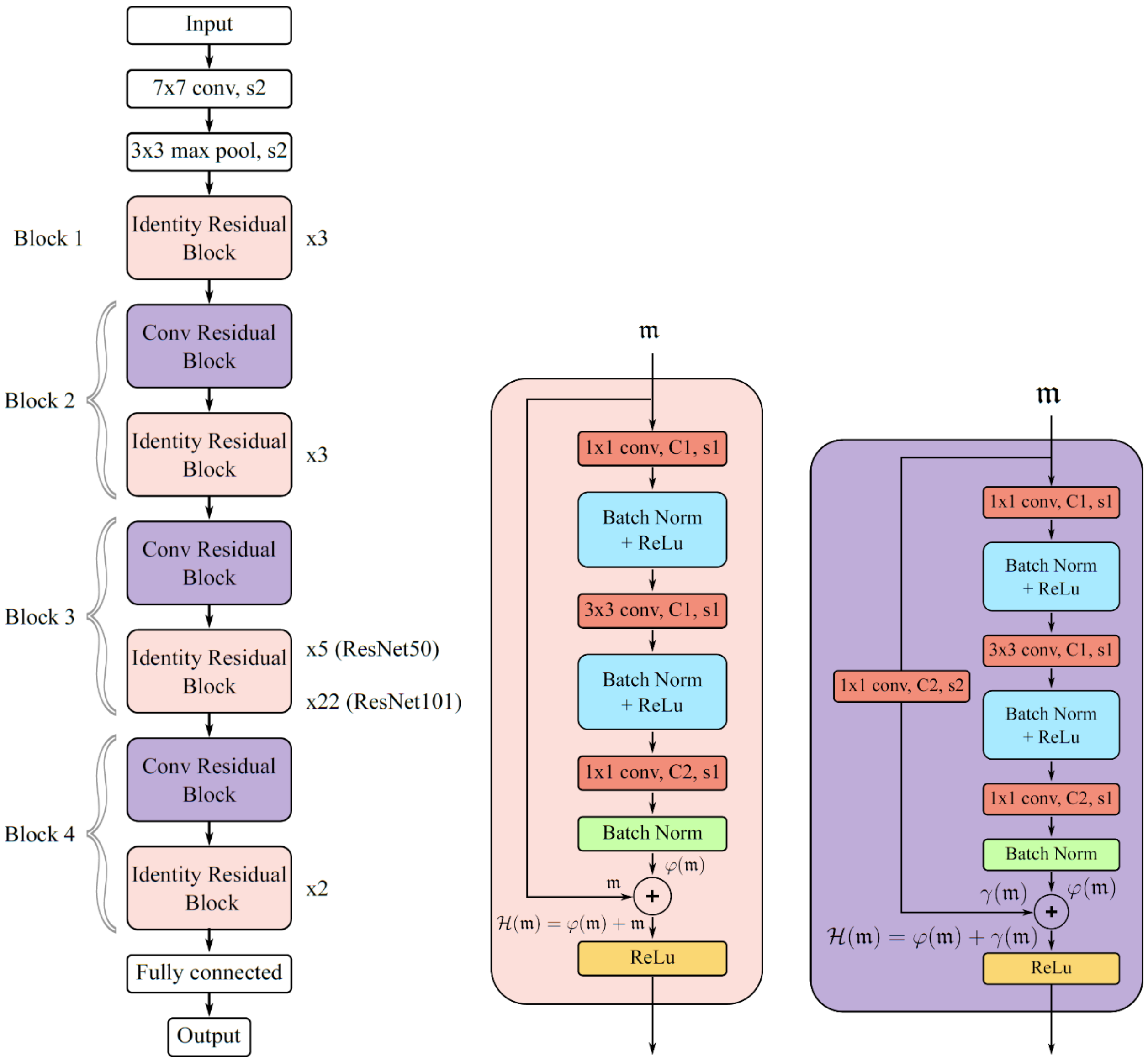


Figure 3. Residual U-Net architecture representation designed for semantic segmentation of weed plants and corn crops.



(a) ResNet architecture

(b) Identity block

(c) Convolutional block

Figure 4. Description of the Backbone adopted for the proposed residual U-Net architecture.(a) Structure of the main blocks of both the ResNet50 and ResNet101.) Identity residual block implemented. This block is used when the size of the feature maps is constant. (c) Convolutional residual block employed for transition steps. This block is needed when the size of feature maps is reduced.

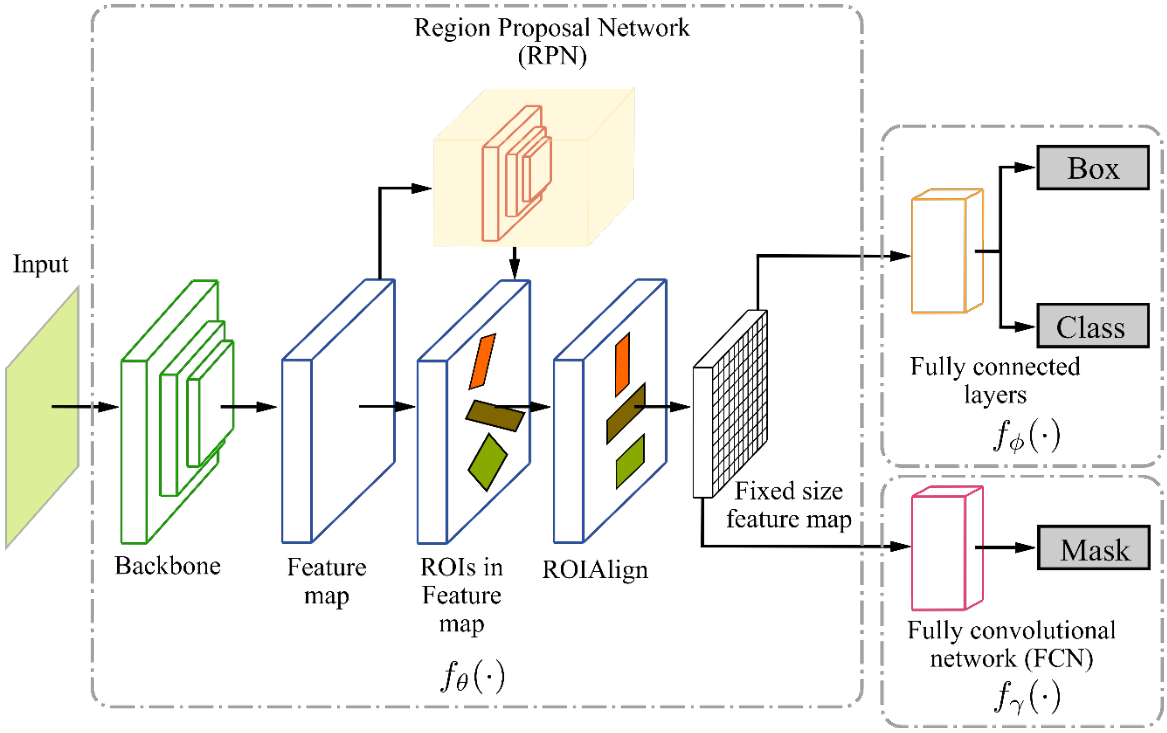


Figure 5. Illustration of the Mask R-CNN architecture for semantic segmentation of weed plants and corn crops.

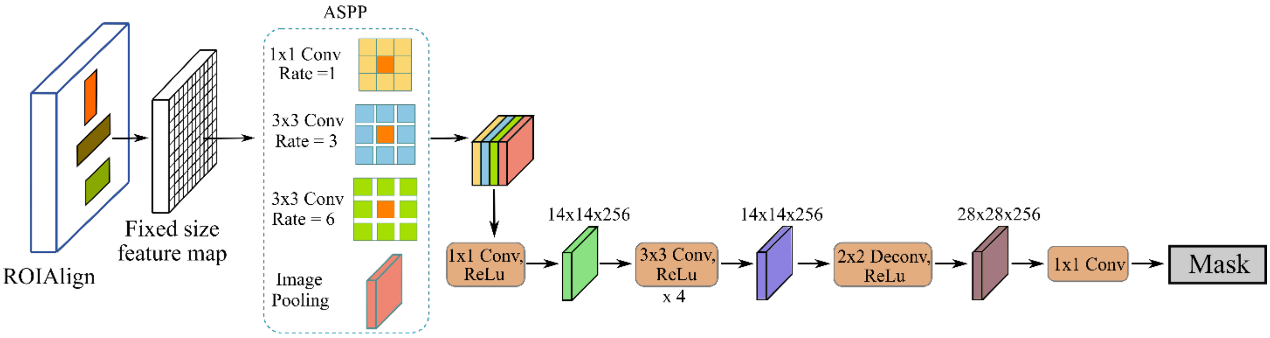


Figure 6. Illustration of the segmentation branch of the Mask R-CNN provided with ASPP to improve the segmentation of corn and weed plants.

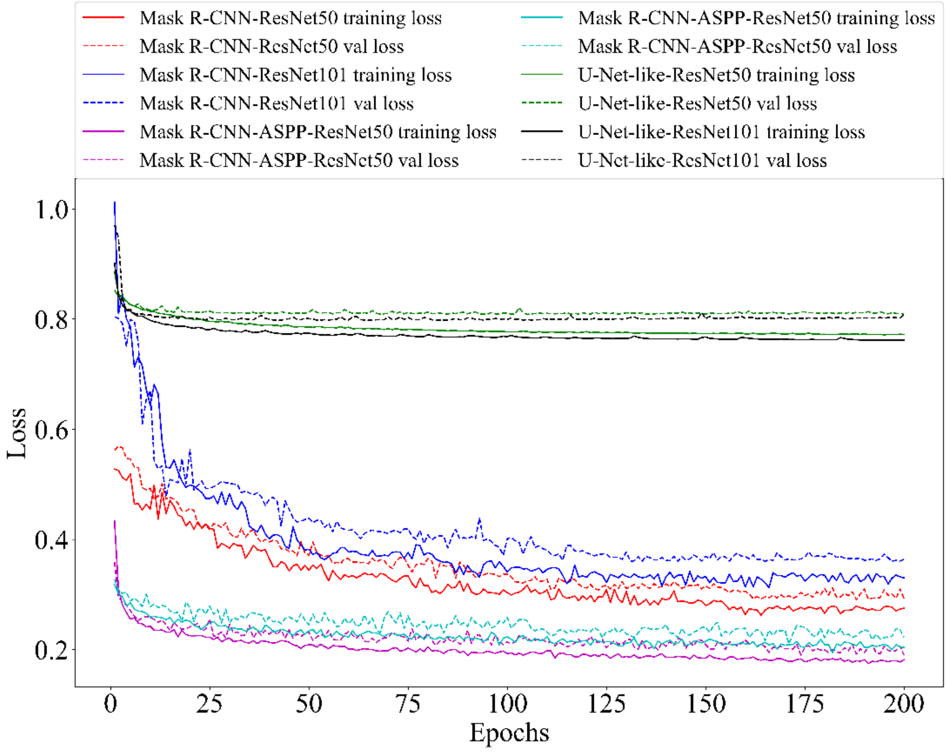


Figure 7. Total loss function behavior of the networks during the training process.

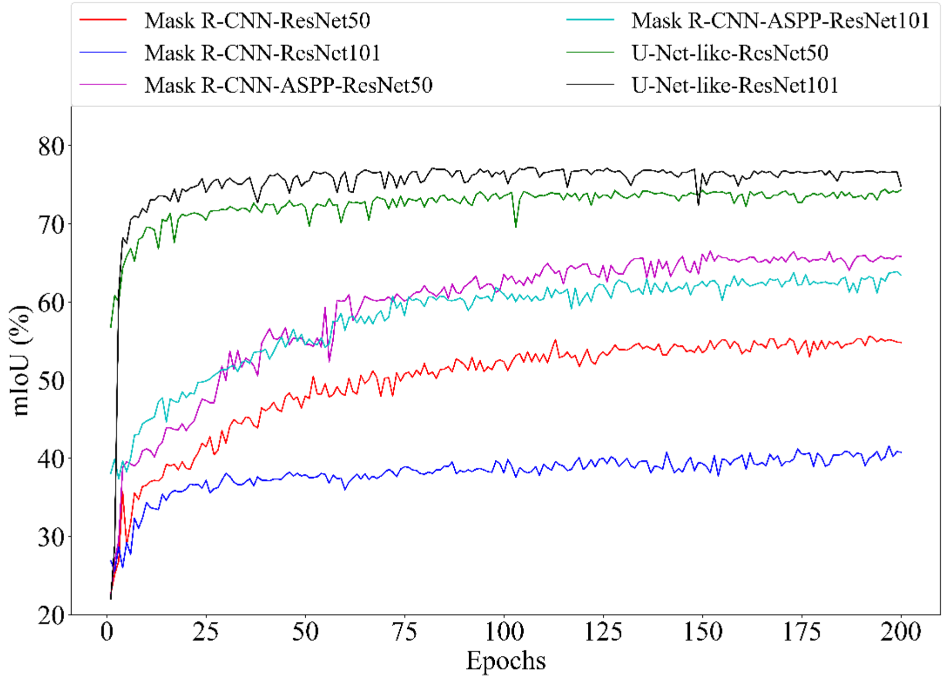
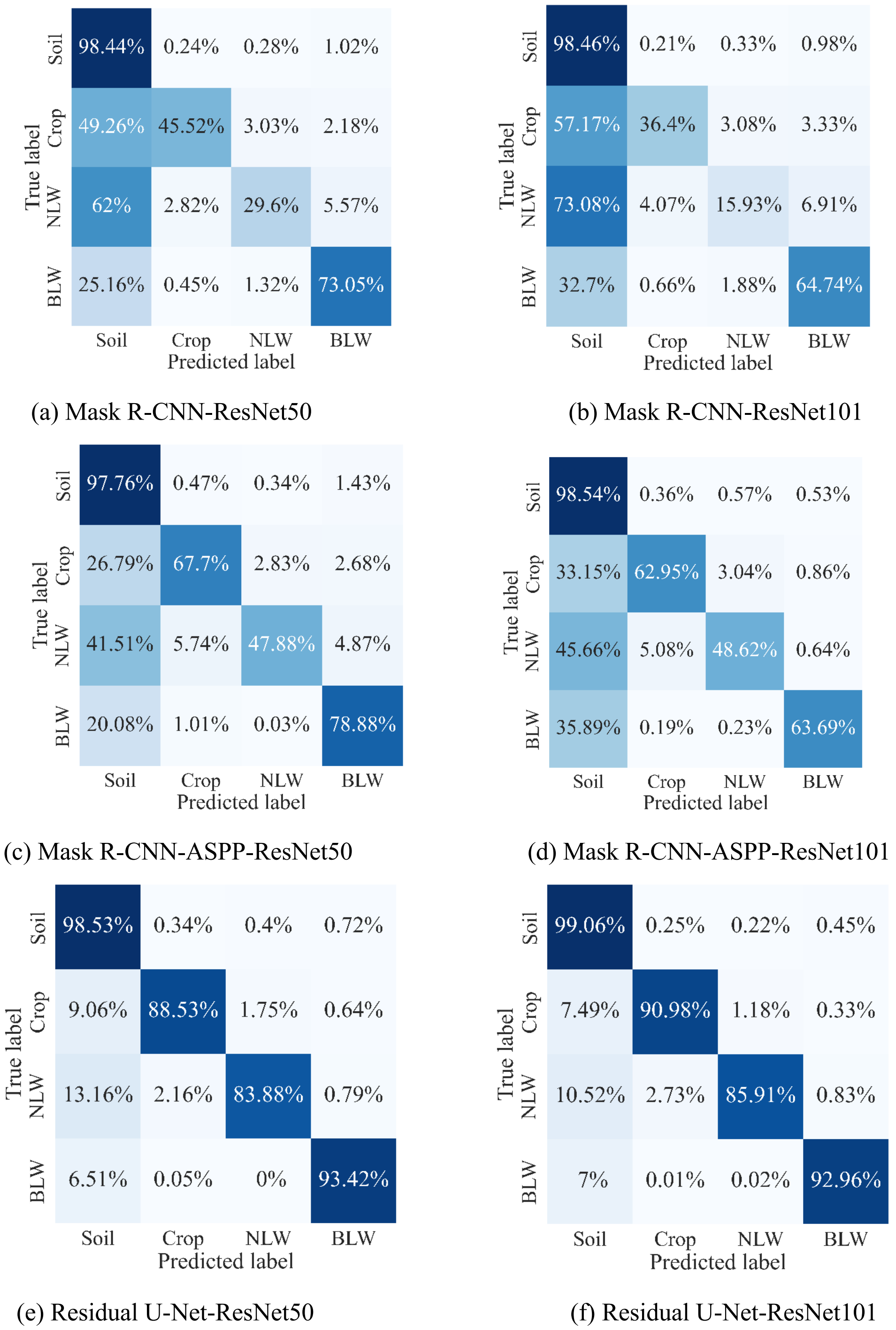


Figure 8. mIoU behavior at each epoch of the networks during the training process.



(a) Mask R-CNN-ResNet50

	Soil	Crop	NLW	BLW
Soil	98.44%	0.24%	0.28%	1.02%
Crop	49.26%	45.52%	3.03%	2.18%
NLW	62%	2.82%	29.6%	5.57%
BLW	25.16%	0.45%	1.32%	73.05%

(b) Mask R-CNN-ResNet101

	Soil	Crop	NLW	BLW
Soil	98.46%	0.21%	0.33%	0.98%
Crop	57.17%	36.4%	3.08%	3.33%
NLW	73.08%	4.07%	15.93%	6.91%
BLW	32.7%	0.66%	1.88%	64.74%

(c) Mask R-CNN-ASPP-ResNet50

	Soil	Crop	NLW	BLW
Soil	97.76%	0.47%	0.34%	1.43%
Crop	26.79%	67.7%	2.83%	2.68%
NLW	41.51%	5.74%	47.88%	4.87%
BLW	20.08%	1.01%	0.03%	78.88%

(d) Mask R-CNN-ASPP-ResNet101

	Soil	Crop	NLW	BLW
Soil	98.54%	0.36%	0.57%	0.53%
Crop	33.15%	62.95%	3.04%	0.86%
NLW	45.66%	5.08%	48.62%	0.64%
BLW	35.89%	0.19%	0.23%	63.69%

(e) Residual U-Net-ResNet50

	Soil	Crop	NLW	BLW
Soil	98.53%	0.34%	0.4%	0.72%
Crop	9.06%	88.53%	1.75%	0.64%
NLW	13.16%	2.16%	83.88%	0.79%
BLW	6.51%	0.05%	0%	93.42%

(f) Residual U-Net-ResNet101

	Soil	Crop	NLW	BLW
Soil	99.06%	0.25%	0.22%	0.45%
Crop	7.49%	90.98%	1.18%	0.33%
NLW	10.52%	2.73%	85.91%	0.83%
BLW	7%	0.01%	0.02%	92.96%

Figure 9. Confusion matrices showing the deep learning models' classification performance. a,b) Confusion matrices corresponding to the R-CNN-based networks. c,d) Confusion matrices corresponding to the R-CNN-ASPP networks. e,f) Confusion matrices for the residual U-Net networks.

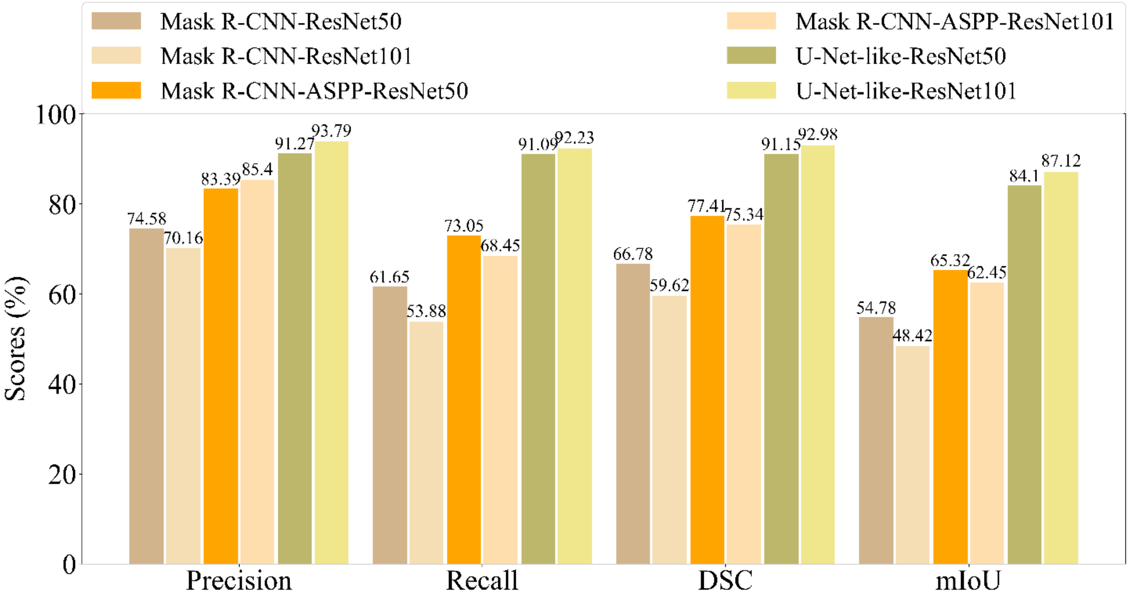


Figure 10. Average performance metrics of the trained networks Mask R-CNN, Mask R-CNN-ASPP, and residual U-Net.

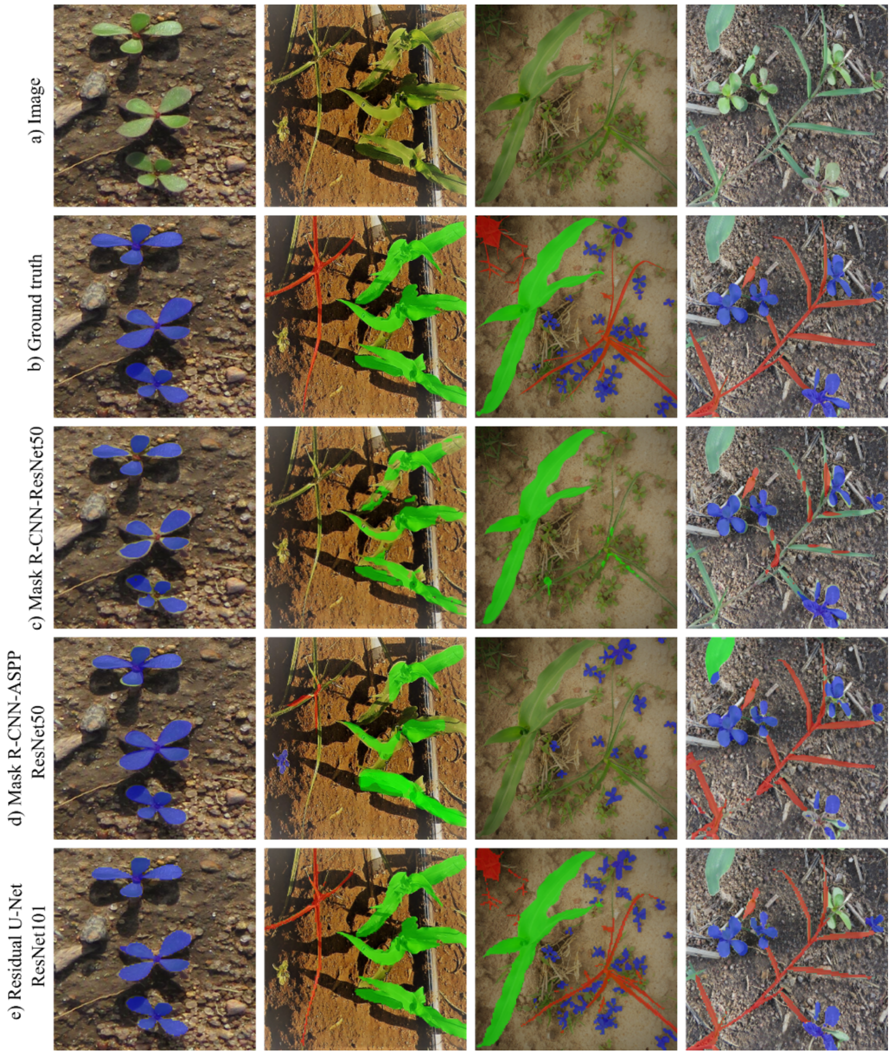


Figure 11. A visual comparison of the segmentation work done by the better three models in each network configuration.

Table 1. Plant species of the experimental dataset and labels.

Class	Scientific name	LBLS	LBLC
Crop	*Zea mays*	18,423	18,423
NLW	*Cynodon dactylon*	5,048	18,636
	Eleusine indica	5,133	
	Digitaria sanguinalis	3,401	
	Cyperus esculentus	5,054	
BLW	*Portulaca oleracea*	5,100	22,622
	Tithonia tubaeformis (Jacq.) Cass.	5,027	
	Amaranthus spinosus	7,388	
	Malva parviflora	5,107	

LBLS, Labels per species; LBLC, Labels per class.

Table 2. Performance evaluation metrics for semantic segmentation models.

Name	Symbol	Expression
Precision	Pr	$\frac{TP}{TP + FP}$
Recall	Re	$\frac{TP}{TP + FN}$
Dice coefficient	DSC	$\frac{2TP}{TP + FP + FN}$
Intersection over union	IoU	$\frac{TP}{TP + FP + FN}$
mean IoU	mIoU	$\frac{1}{C}\sum_{j=1}^{N} IoU_j$

TP, true positive; FP, false positive; TN, true negative; FN, false negative; C, number of classes.

Table 3. Performance of the networks on classifying the classes under study.

Class	Metric	Mask R-CNN		Mask R-CNN-ASPP		Residual U-Net	
		RN50	RN101	RN50	RN101	RN50	RN101
Crop	Pr (%)	66.62	65.85	83.95	86.50	91.15	**93.63**
	Re(%)	45.52	36.41	67.70	62.95	88.53	**90.98**
	DSC (%)	54.08	46.89	74.96	72.87	89.82	**92.29**
	IoU(%)	37.06	30.62	59.94	57.32	81.52	**85.67**
NLW	Pr (%)	51.49	36.84	71.45	71.88	87.02	**91.22**
	Re(%)	29.60	15.93	47.88	48.62	83.88	**85.91**
	DSC (%)	37.59	22.59	57.34	58.00	85.42	**88.49**
	IoU(%)	23.15	12.51	40.19	40.85	74.55	**79.35**
BLW	Pr (%)	83.85	83.38	82.65	89.60	88.32	**91.54**
	Re(%)	73.05	64.74	78.88	63.69	**93.43**	92.96
	DSC (%)	78.08	72.88	80.72	74.46	90.80	**92.24**
	IoU(%)	64.04	57.34	67.67	59.31	83.15	**85.60**
Soil	Pr (%)	96.36	94.59	95.51	93.62	98.61	**98.79**
	Re(%)	98.44	98.47	97.76	98.54	98.53	**99.07**
	DSC (%)	97.39	96.49	96.62	96.01	98.57	**98.93**
	IoU(%)	94.91	93.22	93.47	92.34	97.18	**97.88**

RN50, ResNet50; RN101, ResNet101.

Table 4. Performance of related works upon semantic segmentation of crop/weed in natural environments.

Reference	Model	Classes	DS	NPS	DSC	mIoU
					%	%
Our work	Residual U-Net (ResNet101)	Corn plants	10,200	9	92.98	87.12
		Narrow-leaf weeds				
		Broad-leaf weeds				
Ma *et al.* (2019)	SegNet (VGG16)	Rice seedling weeds	28	-	-	61.8
Khan *et al.* (2020)	CED-Net	Rice	24	2	80.08	71.05
		Sagitaria trifolia				
Zenk *et al.* (2022)	DeepLab V3+ (ResNet50)	Wheat crop	190	-	86.3	77.5
Kamath *et al.* (2022)	PSPNet (ResNet50)	Paddy crop	1,690	-	-	62.43
		Broadleaved weed				
		Sedges				
Picon *et al.* (2022)	PSPNet	Corn plants	1,679	7	25.32	-
		Grass-leaved weeds				
		Broadleaf weeds				

DS: Dataset size (number of images); NPS: Number of plant species; DCS: Dice coefficient, and mIoU: mean Intersection over Union.