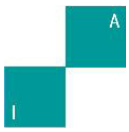
Inteligencia Artificial, 28(75) (2025), 186-198
doi: 10.4114/intartif.vol28iss75pp186-198



INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

R-UNET: Improving performance of rumex detection using unet based selective regions

Saleh Nazal [1,A], Khamael Al-Dulaimi[1,2,B]

[1] Al-Nahrain University, College of Science, Department of Computer Science, Baghdad, Iraq.
[A] b972806@gmail.com
[2] Queensland University of Technology, School of Electrical Engineering and Robotics, Australia.
[B] k.aldulaimi@connect.qut.edu.au

Abstract In particular, Rumex weed detection is regarded as a crucial step in real-world data under many circumstances. The detection task suffers from several issues, such as overlapping weeds, occlusion, varying leaf colour distributions, leaf size and shape, and growth stage. Many machine learning techniques have been proposed to detect weeds in plants. These techniques suffer from locating weed with precise bounding boxes because they may contain multiple bounding boxes in a certain region. Researchers have used the R-CNN based weed identification system, but it continues to have a low detection rate because of the issues mentioned above. In order to detect Rumex weeds under various conditions, particularly overlapping, occlusion, and size, as well as containing multiple bounding boxes, this paper is developed the R-CNN model by using UNet instead of the CNN model to become R-UNet. The proposed model is used due to its novelty of using a UNet classifier with selective regions which boosts the detection capabilities by extracting the most helpful features more effectively than the CNN network. The proposed method uses Intersection over Union (IoU) to assess the detection rate using real-world data. We compare and benchmark the evaluation of the detection performance of this work with different models, including Single-Shot Detector (SSD), hybrid CNNs, AlexNet, and adapted NMS methods. The proposed model yields the highest IoU values compared with other methods.

Keywords: R-UNet, R-CNN, Selective Search, Feature Extraction, Rumex weed, Detection.

1. Introduction

Rumex weed detection is the most important step in smart agriculture. However, it remains a crucial task in real-world data under various conditions in the context of occlusion leaves, different colour of leaves, and overlapping. For dairy farms all across the world, Rumex is the most common immortal weed of its kind [8]. It may spread quickly and yield about 60,000 seeds per year for mature plants. It can be considered the reason for animal health issues [17], reduced dairy productivity and shared crop resources , such as nutrients, area, sunlight, and water. Farmers have used traditional removal based on human effort [17], which is considered laborious, costly, and time-consuming [13]. Various countries have made illegal the use of herbicide treatments that are separated mechanically or manually [19]. Different techniques have been proposed to detect and locate this type of weed from grass regions. For example,To locate and identify weeds in an image with a bounding box, the research of [4] used Region Conventional Neural Networks (R-CNN) and Non-Maximum Suppression (NMS) in conjunction with a number of detectors. However, the occlusion and overlapped weeds utilizing multiple bounding boxes have reduced the performance of the weed detection using the real-world images, making both techniques have poor

ISSN: 1137-3601 (print), 1988-3064 (on-line)

detection performance. [3, 21]. The problems with the majority of weed detection methods are that in order to improve and reliably find object, they had reduced the number of FP and eliminated duplicate detection of a single object [20]. The results of earlier research of Rumex weed detection models have demonstrated that classification and detection have led to low performance measuring and ineffectiveness. They can also be considered costly, time-consuming, and damaging to nature and crops. [17]. Classification and detection processes are the main parts of computer vision. In the agriculture field, the classification step finds the type or class of plants or diseases in an image and the detection step finds where that plant is located in the image. Detection process is a more complex problem than classification because it requires to find the coordinates of the plant in an image. The following points are presented as the innovations of this paper:

1. This work develops a new model architecture to address the challenges of identifying and localising Rumex weed in grassland under numerous circumstances, as well as to improve the performance of Rumex weed identification. The current work is innovative in that it uses the UNET within the R-CNN structure instead of CNN to extract the most valuable features from selected regions and produce multiple bounding boxes after segmenting the image into regions and applying selected features.

2. Compared to current methods, the proposed approach can improve scientific knowledge in the context of agricultural computer vision by developing a real-time localization method and improving detection performance. This model's open resource can be used in combination with the existing methodologies to test the researchers' proposed approaches.

3. This research uses a challenging dataset by Kounalakis et al. (2019) [17]. This dataset is chosen owing to the vast range of grassland conditions, including overlapping, variation in colours, sizes and shapes, and occlusions, which provide significant challenges for CV-based detection.

This paper is oraganized as follows: The relevant work is detailed in Section-2, the proposed method is presented in Section-3, and the evaluation findings are shown in Section-4. Sections-5 and Section-6 give the discussion and conclusion.

2. Related Work

Many methods based DL have been used in weed detection, such as the research of [30] has been proposed DetectNet which based-Deep Convolutional Neural Network (DCNN) model and the GoogLeNet structure. In detecting occluded leaves, the method demonstrated that the outcomes had been improved. The method had trouble finding crops, overlapping leaves, and small weeds. Additionally, it is unable to produce a precise bounding box that fits a plant in its advanced growth stage. Five different approaches have been applied in [12]: Hand-Crafted Features (HCF) with DCNNs; Histogram of Curvature over Scale (HoCS) and HCF-Scale Robust. Compared to the HCF-based DCNN approach, HoCS offered more reliable performance. The findings indicated that those approaches performed poorly even after including a few more occluded images. Graph Convolutional Network (GCN)-ResNet-101 method has been proposed in [15] to identify 6000 mixed of weeds and crops images. The authors first extracted features from ResNet-101 before using semi-supervised GCN to identify crops and weeds. The method's goal was to identify associations between the CNN model's few, labeled features before utilizing Euclidean distances to calculate the separation between each entity. According to the results, the GCN-ResNet-101 technique has outperformed other CNN models. It received high accuracy. However, overfitting and underfitting have happened and resulted in convolutional networks performing poorly on training data. The researchers also noticed that overfitting occurs when there is a small amount of training data. Two CNN-based models, ResNet-50 and Inception-v3, have been proposed by the research of [22] for categorizing weed species from eight distinct regions. Overall, the outcome has shown that ResNet-50's accuracy was marginally superior to Inception-v3's. ResNet-50 obtained high performance because it was made possible by its complicated architecture, which featured a large number of parameters compared to Inception-V3. To recognize the sepium weed in the farm of sugar beet, the CNN-based YOLO-v3 model was combined with real data in [11]. The findings of this method showed that this combination enhanced performance by

7 %. An overlapping case cannot be included in the technique. To extract several plants from an image, the Mask R-CNN model has been utilized in [23]. Each recognized object was given useful information by Mask R-CNN. In [28], the data set which included 14,035 images of 25 various weed species called Weed25. They used the YOLOv3, Faster R-CNN, and YOLOv5 models, and the average accuracy was 91.8 %, 92.15 % , and 92.4 % respectively. The difference in results between the YOLOv3, Faster R-CNN and YOLOv5 models was very small. The CNN algorithm with three models ResNet50, VGG16, and VGG19 was used in [32] with two traditional goal detection algorithms, YOLOv4, and SSD by using to recognize weeds and soybean in the farm under the natural environment. The result showed that VGG19 had effectively identified soybeans and weeds in complex backgrounds comparing to other models. The accuracy of VGG19, which was 5.61 % higher than before optimization, and was 2.24 % higher than the SSD algorithm.

All the methods of weed identification and classification that have been addressed have limitations that must be overcome. These methods have difficulties detecting weed with accurate bounding boxes since they could have several bounding boxes in one area. Also, these are impacted by overlapping, growth stage, illumination and occlusion. Furthermore, some of these methods suffer from insufficient datasets, are time-consuming, and have low precision. To some extent, these methods' problems are discussed in detail in [3].

3. Proposed Model

3.1. Background of R-CNN

CNNs are typically used for image classification, however R-CNN is thought of as an extension of a CNN that i sused for object detection.

Convolutional Neural Network (CNN)

1. CNN is a class of deep neural networks primarily used for feature learning in images.
2. It takes an entire image as input and passes it through layers of convolutional and pooling operations to automatically learn and extract hierarchical features.
3. They are efficient for recognizing patterns, but they do not directly locate objects or regions in images.

Region-based Convolutional Neural Network (R-CNN)

1. R-CNN is designed for object detection and localization in images.
2. It first identifies potential regions of interest in an image using selective search or a similar method.
3. Each region is then passed through a CNN for feature extraction and classification for detection the object
4. R-CNN is more computationally intensive than traditional CNNs due to region proposal generation.

The proposed model is worked on the R-CNN structure it uses the Unet , instead of CNN, to extract the most valuable features from selected regions and produce multiple bounding boxes after segmenting the image into regions and applying selected features. There are three modules in R-CNN detection. The first one is capable of producing region proposals independent of category.These ideas can be used to define the detector's candidate detection set. The second one employs a sizable CNN to extract a fixed-length of feature vectors for each region. The third module employs linear SVMs to classify particular classes [29], as shown in Figura-1

1. Region proposals. Category-independent object proposals [10], [7], selective search [25], and other techniques are some of the numerous ways to create categories or regions. R-CNN is independent of the specific region proposal technique, whereas selective search allows for a controlled comparison with previous detection work.

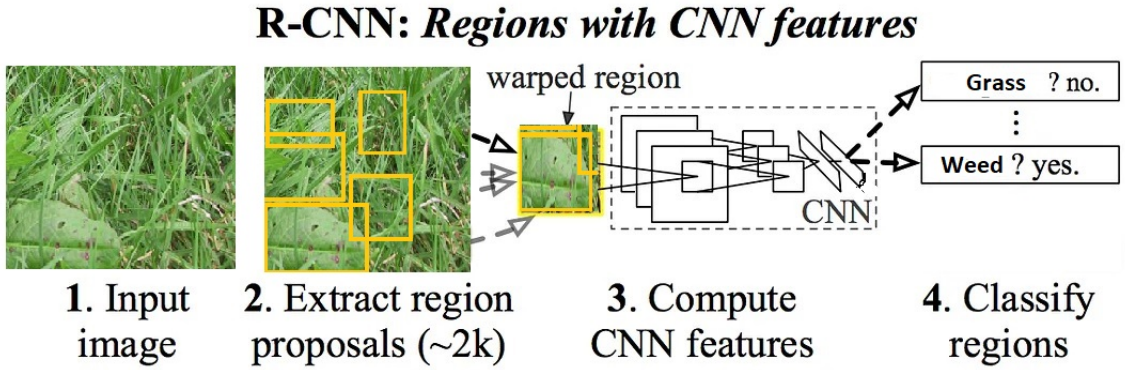


Figura 1: R-CNN:Regions with CNN features

2. Feature extraction.Using the Caffe [14] implementation of the CNN as outlined by Krizhevsky et al. [18], R-CNN's task is used to extract a 4096-feature vector for every proposal region. On a 227×227 RGB image, features are calculated by forward propagating In order to calculate features for a given region, it must convert the picture data in that area into the format that CNN requires.

3. Classifier: A linear SVM model is trained for classification once the features are extracted. We need to classify the objects inside each region.

3.2. Using R-UNet

The novelty of this paper is that it used UNet instead of CNNs to extract features and classification. In the proposed method instead of running UNet model classification on a large number of regions, we can pass the image through a selective search procedure. Then, 2000 proposal regions are first selected from the result, the model is trained, and the classification process is run using the UNet model, as shown Figura -2

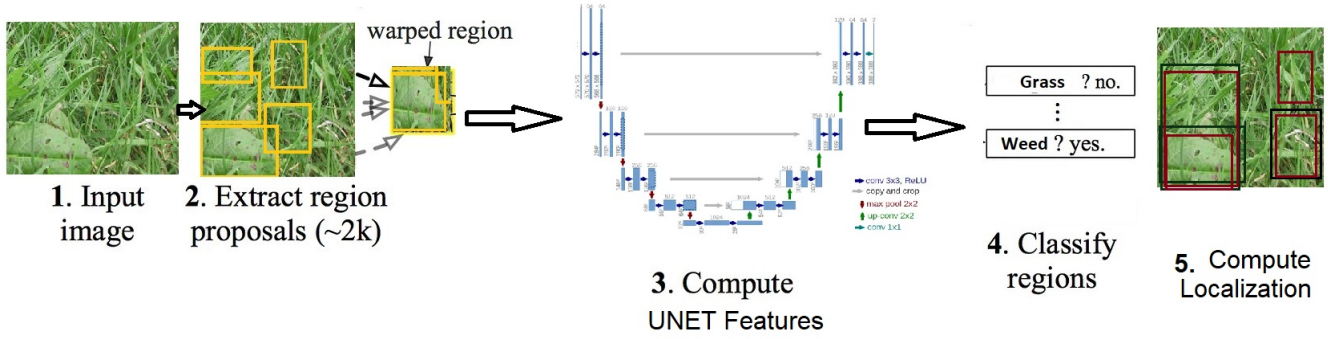


Figura 2: The adapted model R-UNet

3.3. Adapted R-UNet

There proposed method is divided into 4 steps:-

1. Passing the image through selective search and creating proposal regions. The selective search consists of four types of similarity: Fill Similarity; Texture Similarity; Colour Similarity; and Size Similarity. Fill similarity selective search, which evaluates how well two regions fit together, is used

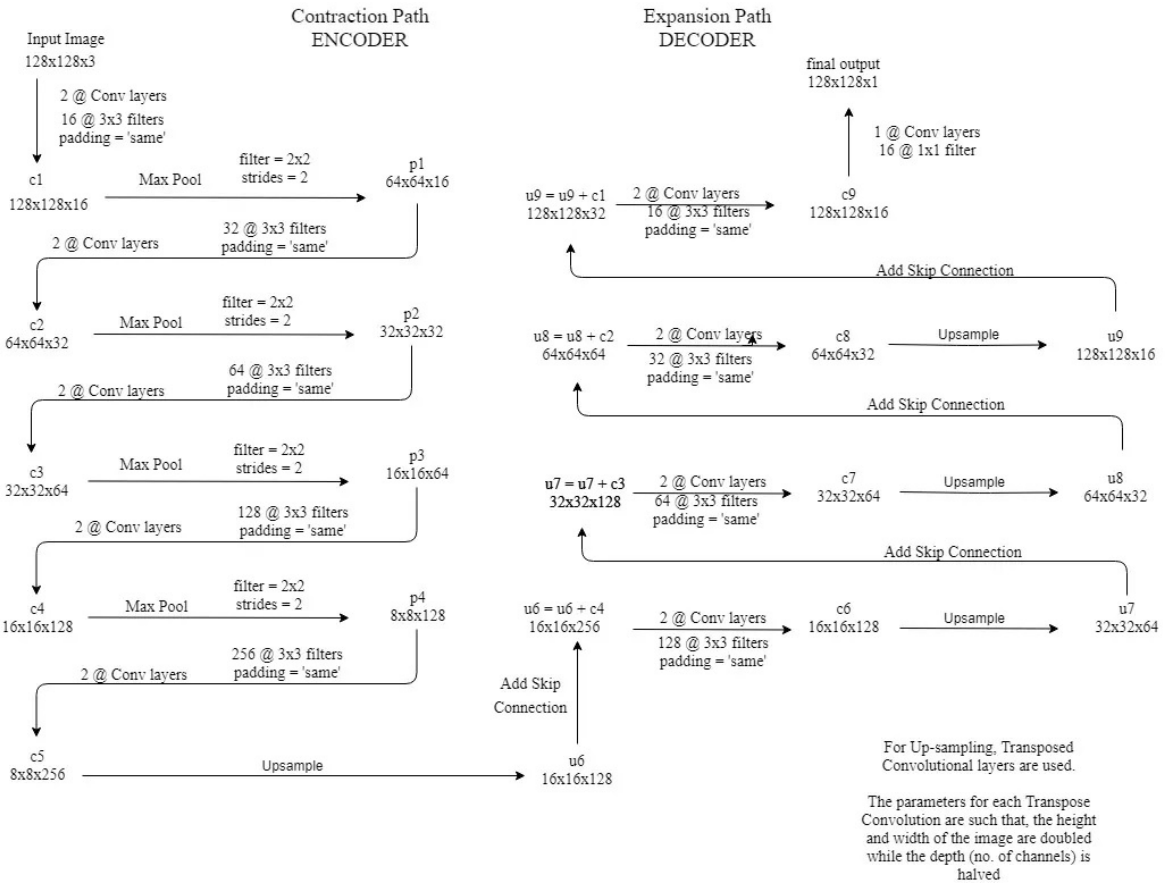


Figura 3: UNet encoder and decoder [24].

in this work. It is not good to merge two areas r_i and r_j that hardly touch one another since they may create an odd region. To keep the measure fast, we can only use the regions' and the containing boxes' sizes. In particular, the small bounding box around r_i and r_j is represented by BB_{ij}. The portion of the image contained in BB_{ij} that is not covered by the areas of r_i and r_j is represented by the $fill(r_i, r_j)$ [25], as written

$$fill(r_i, r_j) = \frac{1 - size(BB_{ij}) - size(r_i) - size(r_j)}{size(im)} \tag{1}$$

2. Calculate IoU (Intersection over Union): The final score is obtained by dividing the overlapping area by the union area. The area of union, or the region that is included by both the predicated and the ground-truth bounding boxes, is the denominator.

3. We need to classify and localize all the weed in the image based on the concept of bounding box. Thus, UNet transfer learning is used based on proposed regions for extracting features and detection . There are two paths in UNet's architecture: 1) The contraction path, sometimes referred to as the encoder, is utilized to record the context of an image.The decoder, additionally referred to as the symmetric expanding path, is used to provide precise localisation using transposed convolutions, whereas the encoder is only a standard stack of convolutional and max pooling layers. Convolutional layers are presented in FCN [24].

The Figura -3 shows two adding Convolution layers that mean two consecutive Convolution Layers are applied. The output tensors of Convolutional Layers are "$c1, c2, ...c9$" parameters, the output

tensors of Max Pooling Layers are " $p1,\ p2,\ p3\ and\ p4$", and the output tensors of up-sampling (transposed convolutional) layers are " $u6,\ u7,\ u8\ and\ u9$". The left side of Figura -3 is the contraction path (Encoder). The image size is gradually decreased while the depth is steadily increased in the encoder, starting from $128 \times 128 \times 3$ to $8 \times 8 \times 256$. The expansion path (Decoder) is shown in the right side of Figura -3. In the decoder, the size of the image is gradually increased, the depth is gradually decreased and the decoder starts from $8 \times 8 \times 256$ to $128 \times 128 \times 1$.
The Decoder gradually applies up-sampling to retrieve "WHERE" the information is located (precise localization). For accurate locations, we can avoid connections at every stage of the decoder by doing concatenation the transposed convolution layers output with the encoder's feature maps at the same level, as shown below:

$$u6 = u6 + c4 \tag{2}$$
$$u7 = u7 + c3 \tag{3}$$
$$u8 = u8 + c2 \tag{4}$$
$$u9 = u9 + c1 \tag{5}$$

The concept of weight updates in the context of gradient descent. Let us say that the output of one neural network given it's parameters is

$$f(x; w) \tag{6}$$

Let us define the loss function as the squared L2 loss (in this case).

$$L(X, y, w) = \frac{1}{2n} \sum_{i=0}^{n} [f(X_i; w) - y_i]^2 \tag{7}$$

In this case the batch size will be denoted as n.Essentially what this means is that we iterate over a finite subset of samples with the size of the subset being equal to your batch-size, and use the gradient normalized under this batch. We do this until we have exhausted every data-point in the dataset. Then the epoch is over. The gradient in this case:

$$\frac{\partial L(X, y, w)}{\partial w} = \frac{1}{n} \sum_{i=0}^{n} [f(X_i; w) - y_i] \frac{\partial f(X_i; w)}{\partial w} \tag{8}$$

Using batch gradient descent normalizes the gradient, so the updates are not as sporadic as if we have used stochastic gradient descent.

4. Results

4.1. Dataset Description

The study of Kounalakis et al. [16] presented the Rumex weed dataset. In the open field of an organic dairy farm in Ancenis, France, the dataset was gathered using a robotic platform. This data set was recorded in a variety of conditions, including occlusion, growth stage, illumination, and overlapping. In addition to 408 XML files for labelling, there are 408 photos with RGB colour JPG images, 256 widths, and 256 heights. In the bounding box, the label of the weed is given as points [16]. Additionally, a file with bounding box information is included in the dataset. Figura-4 provides a selection of real images in different circumstances.

4.2. Image Resizing

To prepare the dataset, image resizing process is used. To train the proposed model, several image sizes are introduced by proceeding with the scaling process using 224×224, 299×299, and 336×336. The optimum results of the proposed model are yielded when the size of the input image is 128×128. As a result, we found that increasing the image size over 128×128 results in high computation.

Figura 4: Sample images from the dataset under different conditions.

Cuadro 1: Programming environment for implementation of the proposed model.

Component	Parameter
Processor	13th Gen Intel(R) Core(TM) i7-1355U 1.70 GHz
RAM	RAM 16.0 GB
Framework	PyCharm Community Edition 2023.3
Library	Keras and TensorFlow model
Windows	OS 11 64-bit operating system, x64-based processor
Programming Language	Python 3.9

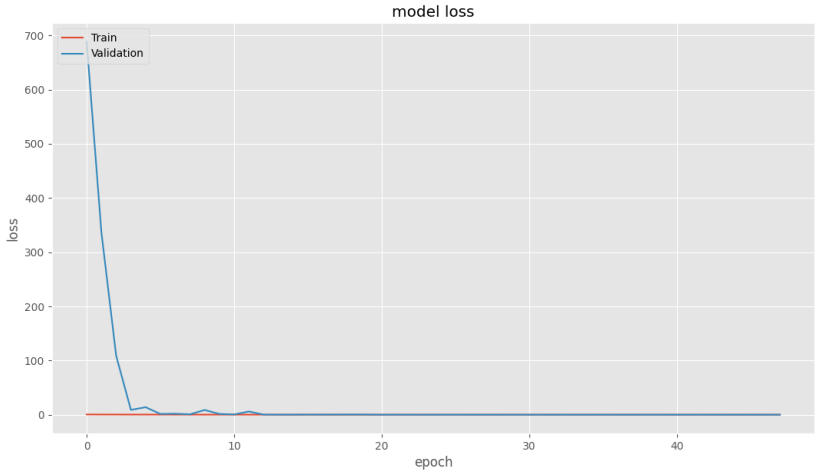
4.3. Data Augmentation

To enhance the quantity of training samples and reduce overfitting, data augmentation is performed. The training data size has been increased by a number of transformations. Each training and validation are augmented for each epoch randomly. Using ImageDataGenerator class, these transformations have been implemented in Python. For implementation, each image is augmented using these parameters:
"horizontal-flip = True,
vertical-flip=True,
rescale=1./255,
rotation-range = 90,
width-shift-range = 0.1,
height-shift-range = 0.1"

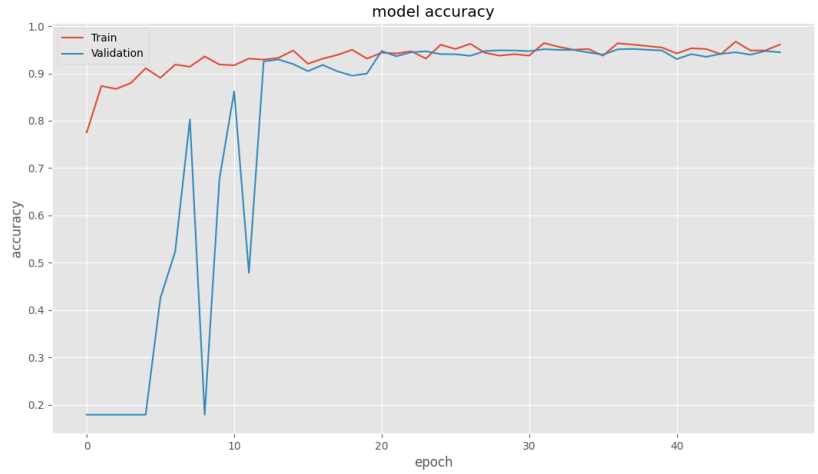
4.4. Experimental Results

Cuadro-1 presents an empirical environment and consists of the objects and the particular hardware and software needed to carry out the experiment.
The procedure of the proposed model is shown in Figura-5. The data is first split into 75 : 25 for training and testing, respectively. Prior to the training process starting, the image size is scaled to 128×128 pixels using image scaling. The training images have been segmented into regions, and 2000 regions have been selected using selective search. In order to increase the training data and create different image conditions, data augmentation is also used. The model is implemented using the Adam optimizer and the binary cross entropy loss function due to the two classes. In an appropriate period of time, the highest accuracy is achieved by the Adam optimizer. RMSprop requires more computation time than Adam optimizer, but it produces the same accuracy. The SGD algorithm yielded the best outcomes with the least amount of training time. SGD will require additional iterations in order to reach the accuracy of

the Adam optimizer, which will increase the computation time. Keras callbacks are used to implement: If the validation loss does not improve after 100 continuous epochs, it is ended early; nonetheless, it is stopped after 54 epochs, as shown in Figura 5 (a) and (b). The weights are only saved if the validation loss is improved. We use a batch size of 32.The input size to the model is $128, 128, 3$ and IoU default value is 0.7. Multiple hyperparameter values are used to train the proposed model. RELU activation is a component of the suggested model's architecture. The outcomes have demonstrated the model's capacity to accurately learns the correlations in a training set, which has improved prediction outcomes. As can be shown in Figure 5 (a), the model learns the problem and achieved zero error. Additionally, a line plot is made that illustrates the mean squared error loss for the train (orange) and validation test (blue) sets over the training epochs. The Figura 5 (a) presents the Rumex weed dataset, which also demonstrate that the model converged in fact quickly and that the performance of the train and validation tests remains consistent. For this task, the model's performance and convergence behaviour indicate that mean squared error provides an appropriate fit for neural network learning.



(a)



(b)

Figura 5: (a)Loss of training and validation with ephocs=50; and (b) Accuracy of training and validation with ephocs=50.

4.5. Evaluation Results

The proposed model's classification results show a high level of accuracy in detecting Rumex weed, at around 94 %. The UNet classifier network based regions and the ideal selective search hold the key achieving high accuracy and an increased the detection rate. In this paper, the IoU metric is employed [1] to compare detection model with other models. Rumex weeds that were captured in challenging real-world scenarios are included in the dataset. The accuracy with which this model can identify an object in an image is calculated by the IoU. Over their union, the predicted and corresponding ground-truth bounding boxes' connection point is calculated [33], as written in Equation-9.

$$IoU = \frac{T \cap P}{T \cup P} \tag{9}$$

Where the target is represented by (T) and the predicted category is represented by (P). When the IoU threshold value is high, the number of the proposed bounding boxes that represents the target object becoming low [33]. Examples of the proposed model's detection results are shown in Figura-6. The red colour box represents the ground truth, while the green colour box refers to the predicted boxes obtained from the proposed model. The proposed model demonstrates its capacity to locate overlapping, occluding, tiny weed, and diseased weed leaves. The majority of techniques are effective against these weeds, although it might be difficult to pinpoint convergence weeds because their leaves overlap and obstruct one another.

This might reduce the True Positive Rate (TPR), which would negatively impact the effectiveness of the detection. The proposed model is demonstrated an outstanding performance on the investigated problems, but it can potentially suffer further problems in some areas that can be taken into account in future research. The proposed model, for instance, is unable to identify some cases of disease because the affected leaves have colours other than green or are tiny. Figura-7 shows some samples that were detected incorrectly. These issues reduce the classification accuracy due to increasing the FPR. This can be resolved by by learning more distinctive features by increasing the training samples.

5. Discussion

In order to address the tiny leaf, overlapping, and occlusion concerns that other approaches have, this research extends R-CNN by employing UNet rather of CNN to create R-UNet. By using the same dataset, the proposed model is compared to recent techniques. It is demonstrated that the proposed model can forecast the amount of weed in grassland and create "better-fitting bounding boxes". The comparison phase compares the proposed model's detection performance to that of other models: AlexNet [26],R-CNN [21] DetectNet, [31], SSD [27], the Hybird CNN model [2], and adaptive non-maximum suppression [4]. Overall, certain techniques are unable to deal with the overlapping or occluded leaves. As the leaf location is close to the edge of image, other techniques performed poorly. In addition, the Hybird CNN model requires training more features to increase its performance. DetectNet has been implemented using the same dataset and has achieved acceptable performance in finding various Rumex over images. However, DetectNet's architecture included a bounding box with a fixed size. As a result of this, Rumex no longer appeared in the scene as full weed. The SSD made many predictions for each object and used IoU to choose the best convergence that matched the target. An adaptive NMS reduced low-confidence proposals and removed duplicate bounding boxes.The SSD completed the entire process using a single network and avoided creating proposal regions, making it computationally faster than competing models. The results of several models are summarized in Cuadro-2. It is demonstrated that the proposed approach performs better than the other compared methods.

6. Conclusion

To enhance the detection performance of weed in the real world, we modify a region-based and selective search model for detecting Rumex weed in grassland environments using R-UNet. The proposed model makes a novel contribution by producing an R-UNet network to identify weeds. The dataset consists of

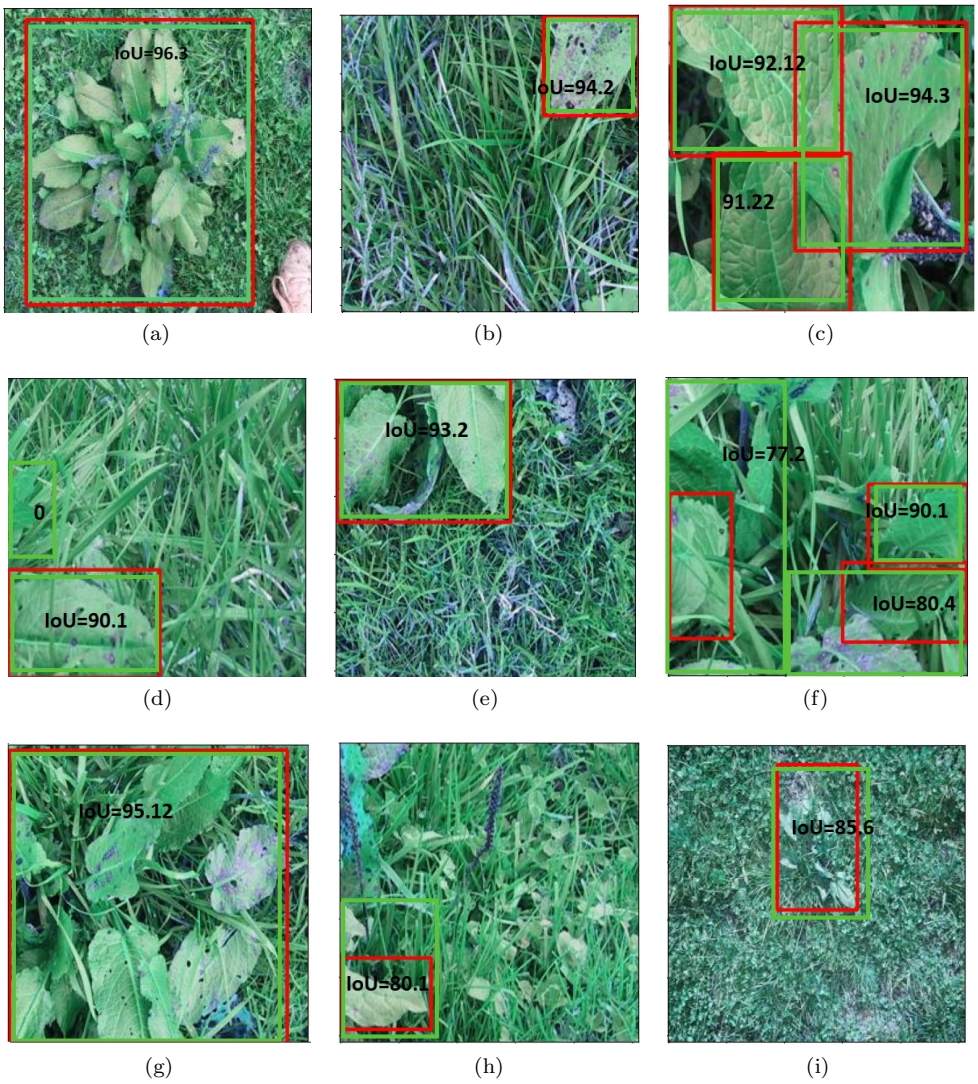
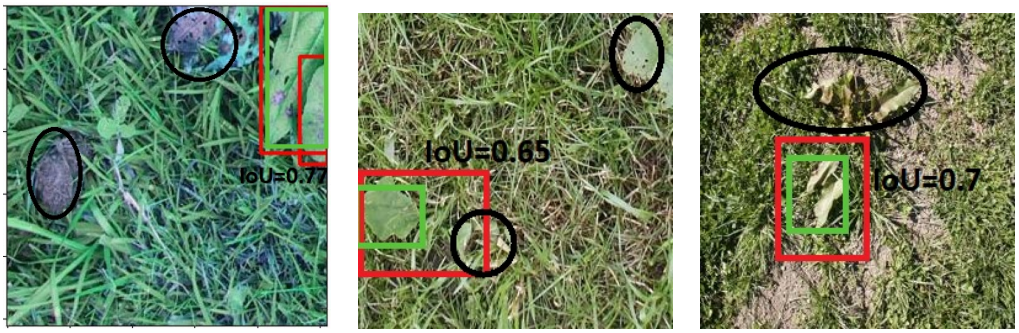


Figura 6: IoU outcome of the proposed model for Rumex weed's tiny, overlapping, occlusion leaves. The green box is the predicated bounding box, whereas the red box is the ground-truth box.

weed images with real-world conditions. In order to overcome the limitations of the previous work, we alter the R-CNN technique in this paper by employing UNet rather than CNN to obtain useful features. This proposed model's key contribution is that it divides an image into regions and uses selective search to store the bounding boxes with the highest scores. Then, we use UNet model to get features and detect the weed. The findings demonstrate that, in contrast to existing models, such as AlexNet and SSD Hybird CNNs model, the proposed model achieves a high detection rate in demanding real-world situations. To increase the robustness of the model, future studies can incorporate datasets on diverse weed species captured in the grassland under varied environmental conditions. Also, ensemble feature fusion from [9], with JSEG [5], and MLP [6] will be used for investigating weed species classification and detection.

Referencias

[1] Shyam Prasad Adhikari, Heechan Yang, and Hyongsuk Kim. Learning semantic graphics using convolutional encoder–decoder network for autonomous weeding in paddy. *Frontiers in plant science*, 10:1404, 2019.



(a) (b) (c)

Figura 7: Some non-green colour cases that were not discovered by the proposed approach were caused by diseases. The ground truth is shown by the red box, the predicted bounding box is represented by the green box, and the undetected weed or leaf is represented by the black oval.

Cuadro 2: Comparison of the proposed model with other models using IoU.

Method	IoU
AlexNet [26]	81.4
SSD [27]	83.5
R-CNN [21]	85.4
Hybird CNN [2]	90.4
Adaptive NMS [4]	91.2
Proposed R-UNet	94.4

[2] Ahmed Husham Al-Badri, Nor Azman Ismail, Khamael Al-Dulaimi, Amjad Rehman, Ibrahim Abunadi, and Saeed Ali Bahaj. Hybrid cnn model for classification of rumex obtusifolius in grassland. *IEEE Access*, 10:90940–90957, 2022.

[3] Ahmed Husham Al-Badri, Nor Azman Ismail, Khamael Al-Dulaimi, Ghalib Ahmed Salman, AR Khan, Aiman Al-Sabaawi, and Md Sah Hj Salam. Classification of weed using machine learning techniques: a review—challenges, current and future potential techniques. *Journal of Plant Diseases and Protection*, 129(4):745–768, 2022.

[4] Ahmed Husham Al-Badri, Nor Azman Ismail, Khamael Al-Dulaimi, Ghalib Ahmed Salman, and Md Sah Hj Salam. Adaptive non-maximum suppression for improving performance of rumex detection. *Expert Systems with Applications*, 219:119634, 2023.

[5] Khamael AL-Dulaimi, Aiman Al-Sabaawi, Rajaa Daami Resen, Jane J. Stephan, and Amani Zwayen. Using adapted jseg algorithm with fuzzy c mean for segmentation and counting of white blood cell and nucleus images. In *6th IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE 2019)*, volume 2019, pages 1–7. 10.1109/CSDE48274.2019.9162402, 2019.

[6] Khamael Al-Dulaimi, Jasmine Banks, Aiman Al-Sabaawi, Kien Nguyen, Vinod Chandran, and Inmaculada Tomeo-Reyes. Classification of hep-2 staining pattern images using adapted multilayer perceptron neural network-based intra-class variation of cell shape. *Sensors*, 23(4):2195, 2023.

[7] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2189–2202, 2012.

[8] Khalid S Alshallash. Emergence and root fragments regeneration of rumex species. *Annals of Agricultural Sciences*, 63(2):129–134, 2018.

[9] Laith Alzubaidi, AL-Dulaimi Khamael, Huda Abdul-Hussain Obeed, Ahmed Saihood, Mohammed A Fadhel, Sabah Abdulazeez Jebur, Yubo Chen, AS Albahri, Jose Santamaría, Ashish Gupta, et al. Meff–a model ensemble feature fusion approach for tackling adversarial attacks in medical imaging. *Intelligent Systems With Applications*, 22:200355, 2024.

[10] Ian Endres and Derek Hoiem. Category independent object proposals. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part V 11*, pages 575–588. Springer, 2010.

[11] Junfeng Gao, Andrew P French, Michael P Pound, Yong He, Tony P Pridmore, and Jan G Pieters. Deep convolutional neural networks for image-based convolvulus sepium detection in sugar beet fields. *Plant Methods*, 16(1):1–12, 2020.

[12] David R Hall. *A rapidly deployable approach for automated visual weed classification without prior species knowledge*. PhD thesis, Queensland University of Technology, 2018.

[13] Carolyn B Hedley and Ian J Yule. Soil water status mapping and two variable-rate irrigation scenarios. *Precision Agriculture*, 10:342–355, 2009.

[14] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678, 2014.

[15] Honghua Jiang, Chuanyin Zhang, Yongliang Qiao, Zhao Zhang, Wenjing Zhang, and Changqing Song. Cnn feature based graph convolutional network for weed and crop recognition in smart farming. *Computers and Electronics in Agriculture*, 174:105450, 2020.

[16] Tsampikos Kounalakis, Michal Jerzy Malinowski, Leandro Chelini, Georgios A Triantafyllidis, and Lazaros Nalpantidis. A robotic system employing deep learning for visual recognition and detection of weeds in grasslands. In *2018 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6. IEEE, 2018.

[17] Tsampikos Kounalakis, Georgios A Triantafyllidis, and Lazaros Nalpantidis. Deep learning-based visual recognition of rumex for robotic precision farming. *Computers and Electronics in Agriculture*, 165:104973, 2019.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[19] Olee Hoi Ying Lam, Marcel Dogotari, Moritz Prüm, Hemang Narendra Vithlani, Corinna Roers, Bethany Melville, Frank Zimmer, and Rolf Becker. An open source workflow for weed mapping in native grassland using unmanned aerial vehicle: Using rumex obtusifolius as a case study. *European Journal of Remote Sensing*, 54(sup1):71–88, 2021.

[20] Damian Mrowca, Marcus Rohrbach, Judy Hoffman, Ronghang Hu, Kate Saenko, and Trevor Darrell. Spatial semantic regularisation for large scale object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2003–2011, 2015.

[21] Saleh Nazal and Khamael Al-Dulaimi. Rumex weed classification using region-convolution neural networks based-colour space information. *Inteligencia Artificial*, 26(72):244–255, 2023.

[22] A Olsen, D Konovalov, B Philippa, P Ridd, JC Wood, J Johns, et al. Deepweeds: a multiclass weed species image dataset for deep learning. *Scientific Reports*, 9(2058), 2019.

[23] Kavir Osorio, Andrés Puerto, Cesar Pedraza, David Jamaica, and Leonardo Rodríguez. A deep learning approach for weed detection in lettuce crops using multispectral images. *AgriEngineering*, 2(3):471–488, 2020.

[24] Twinkle Tiwari and Mukesh Saraswat. A new modified-unet deep learning model for semantic segmentation. *Multimedia Tools and Applications*, 82(3):3605–3625, 2023.

[25] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104:154–171, 2013.

[26] J Valente, M Doldersum, C Roers, and L Kooistra. Detecting rumex obtusifolius weed plants in grasslands from uav rgb imagery using deep learning. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 4, 2019.

[27] Arun Narenthiran Veeranampalayam Sivakumar, Jiating Li, Stephen Scott, Eric Psota, Amit J. Jhala, Joe D Luck, and Yeyin Shi. Comparison of object detection and patch-based classification deep learning models on mid-to late-season weed detection in uav imagery. *Remote Sensing*, 12(13):2136, 2020.

[28] Pei Wang, Yin Tang, Fan Luo, Lihong Wang, Chengsong Li, Qi Niu, and Hui Li. Weed25: A deep learning dataset for weed identification. *Frontiers in Plant Science*, 13:1053329, 2022.

[29] Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. Oriented r-cnn for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3520–3529, 2021.

[30] Jialin Yu, Shaun M. Sharpe, Arnold W. Schumann, and Nathan S. Boyd. Deep learning for image-based weed detection in turfgrass. *European Journal of Agronomy*, 104:78–84, 2019.

[31] Jialin Yu, Shaun M Sharpe, Arnold W Schumann, and Nathan S Boyd. Deep learning for image-based weed detection in turfgrass. *European journal of agronomy*, 104:78–84, 2019.

[32] Xinle Zhang, Jian Cui, Huanjun Liu, Yongqi Han, Hongfu Ai, Chang Dong, Jiaru Zhang, and Yunxiang Chu. Weed identification in soybean seedling stage based on optimized faster r-cnn algorithm. *Agriculture*, 13(1):175, 2023.

[33] Dingfu Zhou, Jin Fang, Xibin Song, Chenye Guan, Junbo Yin, Yuchao Dai, and Ruigang Yang. Iou loss for 2d/3d object detection. In *2019 international conference on 3D vision (3DV)*, pages 85–94. IEEE, 2019.