


Review

Application of Convolutional Neural Networks in Weed Detection and Identification: A Systematic Review

Oscar Leonardo García-Navarrete ^{1,2,*} , Adriana Correa-Guimaraes ¹  and Luis Manuel Navas-Gracia ^{1,*} 

¹ TADRUS Research Group, Department of Agricultural and Forestry Engineering, Universidad de Valladolid, 34004 Palencia, Spain; adriana.correa@uva.es

² Department of Civil and Agricultural Engineering, Universidad Nacional de Colombia, Bogotá 111321, Colombia

* Correspondence: oscarleonardo.garcia@uva.es or olgarcian@unal.edu.co (O.L.G.-N.); luismanuel.navas@uva.es (L.M.N.-G.)

Abstract: Weeds are unwanted and invasive plants that proliferate and compete for resources such as space, water, nutrients, and sunlight, affecting the quality and productivity of the desired crops. Weed detection is crucial for the application of precision agriculture methods and for this purpose machine learning techniques can be used, specifically convolutional neural networks (CNN). This study focuses on the search for CNN architectures used to detect and identify weeds in different crops; 61 articles applying CNN architectures were analyzed during the last five years (2019–2023). The results show the used of different devices to acquire the images for training, such as digital cameras, smartphones, and drone cameras. Additionally, the YOLO family and algorithms are the most widely adopted architectures, followed by VGG, ResNet, Faster R-CNN, AlexNet, and MobileNet, respectively. This study provides an update on CNNs that will serve as a starting point for researchers wishing to implement these weed detection and identification techniques.

Keywords: precision agriculture; weed classification; machine learning; machine vision; image processing; CNN



Citation: García-Navarrete, O.L.; Correa-Guimaraes, A.; Navas-Gracia, L.M. Application of Convolutional Neural Networks in Weed Detection and Identification: A Systematic Review. *Agriculture* **2024**, *14*, 568. <https://doi.org/10.3390/agriculture14040568>

Academic Editors: Ruirui Zhang and Francesco Marinello

Received: 14 January 2024

Revised: 26 March 2024

Accepted: 29 March 2024

Published: 2 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

According to the United Nations, the world population is estimated to reach 9.7 billion inhabitants by 2050 [1]. Against this backdrop, facing the challenge of feeding this growing population with high quality and sustainable products becomes an imperative task. Increasing crop productivity emerges as a measure to address this need. Thus, a strategy that contributes to improving productivity is proper management of weeds, given their direct impact on crop yields. Integrated weed management is essential to preserve agricultural productivity [2]. Plants considered as weeds are fast-growing and actively compete for vital resources, such as space, water, nutrients, and sunlight. This competition not only affects resource availability but also has a negative impact on crop yield and quality [3]. According to [4], damage due to weeds can represent up to 42% of agricultural production.

Currently, diverse weeding techniques are used, such as pre-and post-emergence herbicides, whose application not only generates environmental impacts but also affects the health of the workers who apply them [5]. Mechanical weeding applies mechanized or manual techniques, whose effectiveness in eliminating weeds is not always the desired one, depending on their stage of development [5]. Other weeding alternatives are still under development, or their feasibility has not been fully demonstrated. One is physical weeding using plastic covers [5] and micro-biological weeding involving micro-organisms [6]. Traditional weeding methods present environmental challenges or economic disadvantages, creating the need to explore innovative solutions based on new technologies to increase treatment efficiency. For instance, precision weeding uses image sensors and computational

algorithms to apply herbicides only when weeds are identified [7]. Precision weeding is done by applying herbicides in a variable way, applying a specific amount in the exact place where the weeds are located. Examples of this are commercial developments, such as Blue River Technology from John Deere, WeedSeeker from Trimble, or Bosch Smart Agriculture, among others, which use algorithms based on artificial intelligence (AI) to recognize weeds in the field. In addition to variable rate herbicide application methods, new precision weeding techniques have been introduced commercially, such as the use of lasers to eradicate weeds by burning. For example, LaserWeeding from Carbon Robotics uses AI to identify weeds and eliminate them with a high-power laser in real time; the development of this type of technology reduces environmental impacts and lowers production costs.

Among the AI techniques used in precision weeding is Deep Learning (DL), which is an advanced branch of machine learning (ML) that uses multi-layered artificial neural networks (ANNs) to model and learn complex patterns in data. DL techniques are widely used in agriculture, and their applications are increasing as algorithms are improved [8]. Several studies compiling DL applications in weed detection are presented in the works of [9–12]. Among the different types of DL neural networks is Convolutional Neural Networks (CNN), a type of ANN architecture specially designed to process visual data, such as images and videos. CNNs efficiently detect spatial patterns in digital images by using convolution layers that apply filters to local regions of the input image [9]. These convolution layers allow the network to automatically learn hierarchical and complex features, such as edges, textures, and shapes, instead of relying on predefined features. The basic structure of a CNN model consists of three layers (Figure 1), a convolutional layer, a pooling layer, and a connection layer [10].

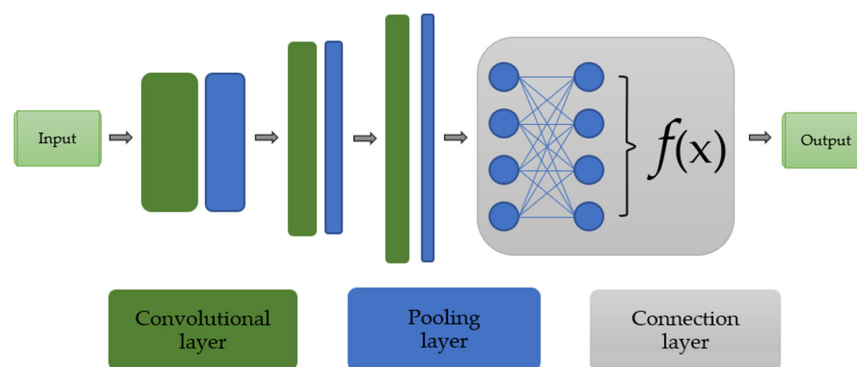


Figure 1. The basic structure of a CNN model.

- The convolutional layer extracts features from the image using mathematical filters; the features can be edges, corners, or alignment patterns, which give the output a feature map that serves as input to the next layer.
- The pooling layer reduces the resolution by reducing the dimension of the feature map in order to minimize the computational cost.
- The connection layer sends the feature maps obtained from the previous layer to the fully connected neural network layer, which contains the activation function used to recognize the final image.

The application of CNNs involves a series of steps: first, data preparation (image acquisition and labeling); second, CNN selection and configuration (hyperparameter tuning); third, CNN training (through GPU processing Graphics Units); fourth, evaluation of CNN performance (usual metrics: average precision (mAP), recall, F1-score, and Confusion Matrix, among others); and fifth, model deployment (real-world applications).

CNNs have gained popularity as an effective image classification method [9]; the success of CNNs is attributed to their ability to process images effectively and extract relevant features automatically, which allows them to generalize new images due to their supervised learning, which has driven their wide use in practical applications and computer

vision research. Although CNNs are efficient in object detection and classification, they present some limitations, e.g., large, labelled datasets are required for training. Lack of training data can lead to deficient performance or over-fitting problems. In addition, computational equipment with high processing capabilities, specifically GPUs, is required, since CNNs have a deep architecture with many parameters, which implies a higher need for processor and memory power.

To improve the efficiency of CNNs, transfer learning (TL) techniques are adopted; these techniques take advantage of the knowledge acquired from pre-trained CNNs to improve their performance in new training [13,14]. TL aims to transfer knowledge from the source domain to the target application by improving its learning performance [15]; this is quite useful when the dataset for the target application is small or limited, as the pre-trained model can provide useful representations of the data and speed up the training process.

The application of CNNs in agriculture, specifically for weed identification, has significantly improved traditional weeding, which employs classical computer vision, where the analysis is carried out pixel by pixel, and is computationally expensive in processing, especially in execution time, so the use of CNNs in weed identification has improved the detection, localization and recognition of weeds, besides, being faster in achieving real-time applications [8,9,16]. Although, in practice, weed detection faces several problems, such as similarities in color, textures, and shapes, as well as occlusion effects and variations in lighting environments, CNNs supported by large-scale datasets have shown great robustness to biological variability and diverse imaging conditions [13], leading to more accurate classification or detection [17,18], which allows for much more accurate and efficient automation of weeding processes [7,16].

One of the examples of using TL with CNN is AlexNet, which was trained on the ImageNet dataset [19]. In agriculture, the TL approach has been implemented in weed detection and classification, helping to minimize the need for large-scale image data collection and reduce the computational costs associated with the training hours of a new CNN model [20–22]. In addition to the use of TL in agriculture, some generative adversarial network (GAN) techniques have been applied to generate artificial images in order to augment the training set with TL [23]. The evolution of CNNs has been marked by significant advances in terms of architecture, training techniques, efficiency, and applications, starting from the need to implement fast solutions in image analysis; some of the most used CNN architectures in weed detection are mentioned below:

- AlexNet: Developed by [19] in 2012, consists of eight convolutional layers and five fully connected layers, with the deep architecture allowing learning of complex hierarchical features from images. It introduced the effective use of Rectified Linear Unit (ReLU) activation functions, which helped mitigate the gradient fading problem and accelerated training. It won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competition, demonstrating the impact of CNNs in computer vision, and is considered one of the most influential architectures in the field. Its main advantage is efficiency in training by using GPUs, allowing faster and more efficient training compared to previous methods. Among its disadvantages is that it requires large datasets to learn the complex hierarchical features of the image; therefore, it is also computationally demanding to train and run.
- GoogLeNet (Inception): Developed in 2014 by the Google Research team, it introduced the concept of Inception modules with multiple filter sizes in parallel [24]. The modules are complex network structures with multiple convolution operations operating in parallel with different filter sizes. These modules allow the capturing of features at different scales and hierarchies, significantly improving the model's ability to recognize complex image patterns. One of the most significant advantages is its high performance with fewer parameters than other architectures, making it computationally efficient. In addition, this architecture captures complex features at different scales, which improves its generalization capability and helps to reduce the risk of

overfitting in large datasets. As for drawbacks, Inception modules can be complex to implement and understand, making them difficult to use for some developers, and they require large amounts of memory to train and execute, due to their depth and complexity. Although the original GoogLeNet architecture (Inception v1) has been widely used, there have been several versions incorporating improvements; Inception v2 and v3 integrate Batch Normalization techniques, Inception v4 incorporates larger and more complex modules, as well as residual connections inspired by ResNet, and opens the door to versions such as Inception-ResNet and Inception-ResNet v2, improving efficiency and accuracy.

- VGG: Developed in 2014 by the Visual Graphics Group (VGG) at Oxford University, it is known for its simplicity and depth [25]. It is characterized by its depth and uniform structure, mainly composed of 3×3 convolutional layers and 2×2 pooling layers, followed by fully connected layers. The VGG architecture can have different configurations, from VGG16 (with 16 layers) to VGG19 (with 19 layers), depending on the number of convolutional layers. As an advantage of VGG, this architecture allows efficient feature extraction at different scales and abstraction levels due to its simple structure of 3×3 convolutional layers, and its easy implementation due to its uniform and symmetric architecture. Among the disadvantages of VGG is the computational cost in training and execution, due to the depth and number of parameters; also, the large number of parameters in VGG models can increase the risk of overfitting, especially in small datasets.
- ResNet (Residual Network): Developed by [26] in 2015, this architecture was highlighted as using residual blocks, which allow the training of very deep networks. The residual blocks are based on facilitating the flow of information by creating a direct connection path (known as “skip connection” or “shortcut connection”) that prevents the gradient from disappearing during training. ResNet has undergone several versions where the number of layers is increased to improve prediction. The main ones are: ResNet-50, ResNet-101, and ResNet-152, variants with different depths, having 50, 101 and 152 layers, respectively. Their main advantage is based on the ability to train deeper networks by their residual connections, which decreases the degradation in performance, in addition to minimizing overfitting and improving the generalization capability of the model, which allows taking full advantage of the power of deep learning. Despite its efficiency, a disadvantage of ResNet is a higher requirement of significant resources in terms of memory and computational power, especially when training very deep architectures.
- Fast R-CNN: an object detection model introduced in 2015, significantly improving its predecessor’s speed and efficiency, is the R-CNN (Region-Based Convolutional Neural Network) model. It was developed by the Microsoft Research group to address the speed and computational efficiency limitations associated with R-CNN, providing a faster and more practical solution for object detection in images [27]. Fast R-CNN uses a pre-trained CNN network to extract features from the input image. Then, regions of interest (ROIs) are generated using a technique called region proposal (e.g., using algorithms such as Selective Search), and these regions are transformed into a fixed region to be input to the CNN network. Finally, bounding boxes of these proposed regions are classified and fitted using classification and regression layers, respectively [27]. The advantage of Fast R-CNN is the runtime efficiency of using a single CNN network to extract features and perform region classification rather than passing each proposed region through a separate network. A disadvantage is that it may have difficulty detecting small objects, or in cases of object overlap. Likewise, this architecture has variants, such as Mask R-CNN, which adds an additional branch to the network to perform semantic segmentation of objects in the image, object detection, and classification.
- DenseNet (Densely Connected Convolutional Network): In 2017, proposed by [28], this is notable for its densely connected structure, where each layer is directly con-

- connected to all subsequent layers. This dense connectivity can potentially improve information flow and mitigate the problem of gradient fading. It has influenced the design of subsequent architectures and continues to be a popular choice in research and practical implementation in computer vision tasks. Although DenseNet has been used primarily in its original form since its introduction, there have been some proposed extensions and variants, such as DenseNet 121, 169, and 201, each with a different depth. These numbers represent the total number of layers in the network, including convolutional layers, pooling layers, fully connected layers, and normalization layers. The main advantage of DenseNet is the direct flow of information from the input layers to the output layers, facilitating the learning of complex features and the propagation of gradients through the network. In addition, having a direct data flow mitigates the problem of gradient fading, which facilitates the training of deeper networks. As a disadvantage, it has a higher computational cost, mainly in memory, due to its dense connections requiring more training and inference computations.
- MobileNet: Proposed in 2017 by [29], it is specially designed for implementations on mobile devices and uses lightweight and efficient operations to balance performance and resource consumption. The main feature of MobileNet is its ability to strike a balance between network accuracy and computational efficiency through a series of building blocks called “Depthwise Separable Convolution” that significantly reduce the number of parameters and the amount of computation. The building blocks divide the standard convolution into two separate stages: a depthwise convolution followed by a pointwise convolution; this allows for drastically reduced computational cost without sacrificing too much accuracy [29]. The advantages of MobileNet are computational efficiency and low resource consumption, which makes it ideal for running on resource-constrained devices, such as cell phones and IoT devices. Its main disadvantage is lower accuracy compared to other larger and more complex architectures in certain computer vision tasks. Attempts have been made to improve the architecture by obtaining versions such as MobileNetV2 and MobileNetV3, which improve accuracy and performance.
 - YOLO (You Only Look Once): Developed in 2016 by [30], it is a fast and efficient object detection architecture, as it approaches this task as a regression problem; instead of a separate classification for each region, this feature allows several versions from YOLOv1 up to YOLOv8 in 2023. Starting with the fifth version, released in 2020, known as YOLOv5, this was built on PyTorch [31], maintaining the original YOLO approach of dividing the image into a grid and predicting bounding boxes with class probabilities for each cell. The overall architecture includes convolutional layers, attention layers, and other modern techniques; it is important to mention that this version was developed by the Ultralytics team, not by the original authors. In 2022, the YOLOv6 and YOLOv7 versions were developed, presenting improvements in their architecture and training scheme, and improving object detection accuracy without increasing the cost of inference, a concept known as “trainable feature bags” [32]. Finally, in 2023, YOLOv8 was presented; its improvements included new features, better performance, flexibility, and efficiency. Additionally, it includes improvements for detection, segmentation, pose estimation, tracking, and classification [33].

The following is a systematic review of the latest work on the detection and identification of weeds with CNN techniques to serve as a starting point for researchers wishing to implement this technique; for this purpose, the following sections are introduced: (a) Methods, using the PRISMA statement; (b) Results: developing the research question and producing statistics for the literature analysis, of the CNNs used for the detection and identification of weeds, of the sources of image acquisition for training and the study species in the articles reviewed; (c) Discussion: analysis of the articles found based on the proposed objectives; and (d) Conclusions: conclusions and proposals for future work are developed.

2. Methods

In this study, a systematic review was carried out to identify and analyze the scientific literature published on weed detection using CNNs. The guidelines of the PRISMA statement [34] were followed in this review.

2.1. Research Question and Review Objectives

1. Research question: What are the Convolutional Neural Network (CNN) architectures most frequently used for weed detection and what are the most commonly used image acquisition sources for training these CNNs??
2. Main Objective: Analyze the different CNN architectures used for weed detection and identify the sources of image acquisition for CNN training.
3. Specific Objectives:
 - To identify and analyze the most commonly used CNN architectures for weed detection in different crops.
 - To determine the image acquisition sources most commonly used in CNN training for weed identification in different forms of production.

2.2. Sources of Information

The following databases were used for the systematic search: Web of Science and Scopus.

2.3. Search for Keywords

A primary search was carried out to establish the relevant words for the systematic search; for this, the Scopus database was used with the words “weeds detection deep learning,” establishing the “search within all fields,” (ALL (weeds AND detection AND deep AND learning)); in this search, 6096 results were obtained. With the filtering options in “Filter by keyword,” we found the five most commonly used keywords with their number of matches: “Deep Learning” (1800), “Machine Learning” (988), “Remote Sensing” (704), “Crops” (647), and “Convolutional Neural Networks” (638). The initial search covered the topics of importance for this systematic review and took “Weed detection,” “Deep Learning,” and “Convolutional Neural Networks” as keywords for the search.

2.4. Inclusion and Exclusion Criteria

The following inclusion and exclusion criteria were used and implemented through the filters of each database:

1. The search field is selected where the search is directed through titles, abstracts, and keywords, among others; this is specific to each database:
 - In Scopus, “search within Article title, Abstract, Keywords” was established.
 - In Web of Science, the search was established in “Topic”; this includes title, abstract, author keywords and keywords plus.
2. The date range of the search is the last five years, from 2019 to 2023.
3. Document type: “Document type: Article”.
4. Excluded are reviews, book chapters, narrative articles, conference or congress articles, unofficial notes or communications, and studies from other areas, such as social, human, biological, chemical, legislative, social and economic impacts.
5. Language: “English Language”.

2.5. Search String in Bibliographic Databases

The search equation is established by restrictively connecting all the results containing the keywords “weeds detection” AND “Deep learning” AND “Convolutional Neural Networks.” With this, the search equation is established according to each platform:

- Scopus: TITLE-ABS-KEY (“weed detection” AND “deep learning” AND “Convolutional Neural Networks”)

- WOS: TS = (“weed detection” AND “deep learning” AND “Convolutional Neural Networks”)

2.6. Initial Search Results

1. Initial records: Scopus 104 and WOS 40. Initial results obtained, 144.
2. Records eliminated by exclusion criteria: 65 results eliminated and 79 results remained, 61 from Scopus and 18 from WOS.

2.7. Duplicates and Screening

The free access tool Zotero 6.0.30 was used to eliminate duplicate results, and 14 duplicate results were eliminated, obtaining a total of 65 results.

2.8. Additional Records

16 articles obtained from reading book chapters and reviews were added to the results, obtaining 81 results.

2.9. Records Excluded

A total of 20 records were excluded because they do not meet the objective of this review or cannot be accessed.

Finally, 61 articles that meet the established criteria were gathered and analyzed. Figure 2 illustrates the process in a flow chart using the PRISMA methodology.

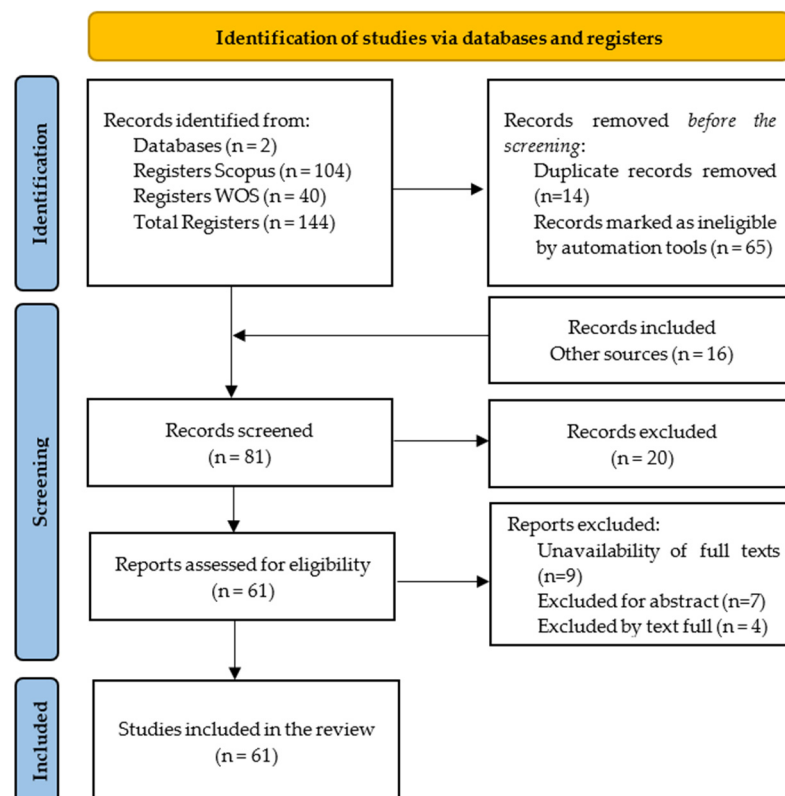


Figure 2. Flow chart illustrating the number of articles included in the systematic review according to the PRISMA process.

3. Results

3.1. Literature Analysis

A detailed analysis of the 61 bibliographic articles was conducted. The analysis indicated that, in the last two years, there has been a growth in the number of articles published on weed detection (Figure 3). The increasing amount of research in this area

is mainly due to the development of new and more efficient CNN architectures, the increase in the processing capacity of computers, and the reduction in the price of cameras and GPU.

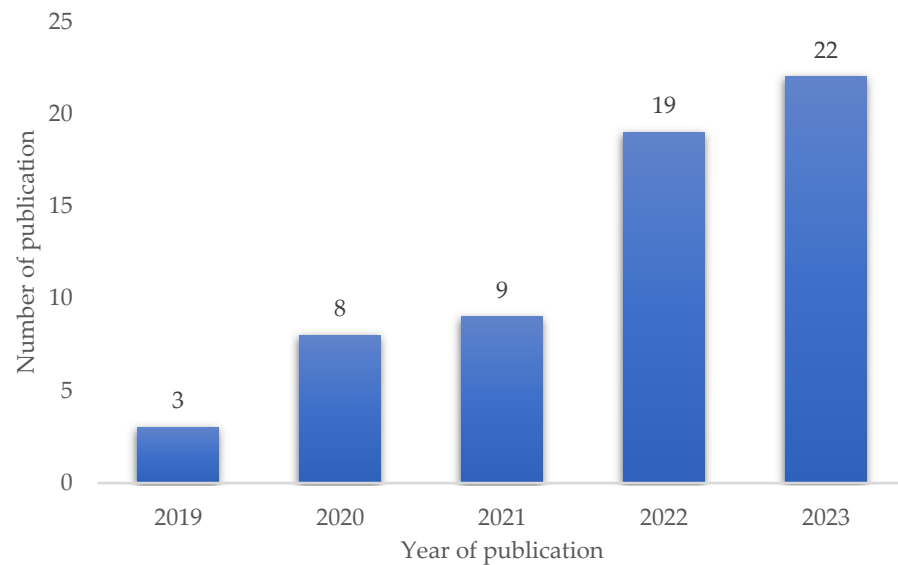


Figure 3. Number of publications for specific years.

3.2. Source of Images for the Training of the CNNs

In the review of the selected articles, it was found that the authors used different types of sources for the acquisition of the images used in the training and validation of the CNNs, such as digital cameras, professional Reflex type, industrial high-speed and low-cost cameras, such as those using Raspberry cards. UAVs had various types of cameras, both RGB and multi-spectral. Smartphones with high-resolution cameras were also used. In addition, it was found that some researchers did not acquire images and used free databases or previous works. Figure 4 shows the number of publications according to the source used, where 49.2% used digital cameras, 29.5% used UAV as a mean of acquisition, 11.5% used smartphones, and 9.8% used already built datasets.

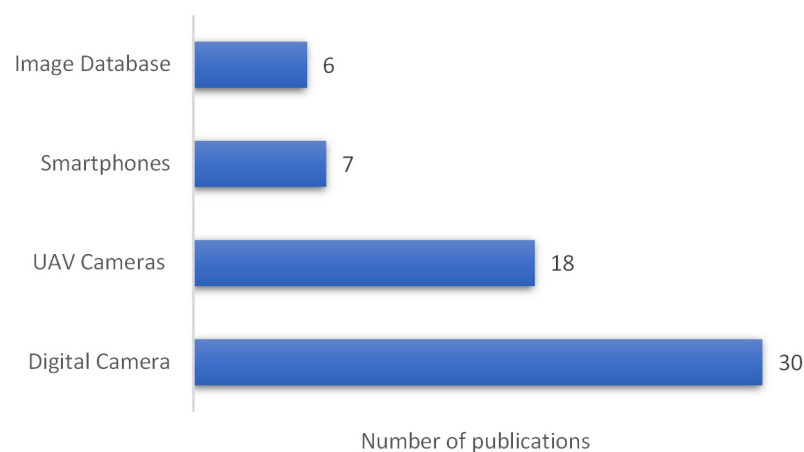


Figure 4. Number of publications by image source for training of CNNs.

3.3. CNN Architecture Used

Table 1 summarizes relevant information extracted from the selected reviewed studies on CNN architectures.

Table 1. Summary of articles describing CNN architecture and source of image.

No.	Reference	CNN Architecture	Source of Image
1	[35]	Specifically designed CNN	Database
2	[36]	Design-specific CNN with SVM	Digital Camera
3	[37]	AlexNet, GoogLeNet and VGGNet	Digital Camera
4	[38]	SegNet, UNet, VGG16 and ResNet-50.	Digital Camera
5	[39]	CRowNet is based on SegNet and the Hough transform	UAV multi-spectral camera
6	[40]	YOLOv3 and YOLOv3-Tiny	Digital Camera
7	[41]	ResNet50, VGG16, VGG19, Xception and MobileNetV2.	Digital Camera
8	[42]	Mask R-CNN	Digital Camera
9	[43]	YOLOv3, Mask R-CNN, and CNN with SVM-HOG (histograms of oriented gradients)	UAV multi-spectral camera
10	[44]	YOLO-WEED, based on YOLOv3	UAV Camera
11	[45]	Faster R-CNN and Single Shot Detector (SSD)	UAV Camera
12	[46]	CNN-LVQ-specific design based on Learning Vector Quantization (LVQ)	UAV Camera
13	[47]	YOLOv3, YOLOv3-Tiny and YOLOv3-Tiny-PRN	Digital Camera
14	[48]	Faster R-CNN and ResNet50	Digital Camera
15	[49]	Detectron2, EfficientDet, YOLOv5, and Faster R-CNN.	Digital Camera
16	[50]	Faster R-CNN, ResNet-101, VGG16 and Yolov3	UAV Camera
17	[51]	VGG-Beet based on VGG16	UAV multi-spectral camera
18	[52]	YOLOv5 and Classic K-Nearest Neighbors, Random Forest, and Decision Algorithms tree	UAV multi-spectral camera
19	[53]	Faster R-CNN and Mask R-CNN	Digital Camera
20	[54]	Faster R-CNN and VGG16	Digital Camera
21	[55]	VGG16, ResNet-50 and Inception-V3	Digital Camera
22	[56]	AlexNet vs. VGG-16	UAV Camera
23	[57]	YOLOv4, YOLO- sesame.	Digital Camera
24	[58]	VGG16 and ResNet16	Digital Camera
25	[59]	YOLOv3-Tiny	Digital Camera
26	[60]	YOLOv5	Digital Camera
27	[61]	Faster R-CNN, SSD, YOLOv3, YOLOv3-tiny and YOLOv4-tiny	Smartphone
28	[62]	Faster R-CNN and ResNet	Smartphone
29	[63]	UNet based on ResNet50	Digital Camera
30	[64]	MobileNetV2, ResNet50	UAV Camera
31	[65]	YOLO-v3 and faster R-CNN	Digital Camera

Table 1. Cont.

No.	Reference	CNN Architecture	Source of Image
32	[66]	Faster R-CNN, ResNet-101, YOLOv4, SSD-Inception-v2, MobileNet, ResNet-50, EfficientDet and CenterNet	Database
33	[67]	SSD-MobileNet, SSD-InceptionV2, Faster R-CNN, CenterNet, EfficientDet, RetinaNet and YOLOv4	Database
34	[68]	YOLOv4 and Faster R-CNN.	Digital Camera
35	[69]	Mask R-CNN and GAN	Digital Camera
36	[70]	Alexnet, GoogLeNet, InceptionV3, Xception	Smartphone
37	[71]	VGGNet (16 and 19), GoogLeNet (Inception V3 and V4) and MobileNet (V1 and V2)	Digital Camera
38	[72]	VGG, Resnet, DenseNet, ShuffleNet, MobileNet, EfficientNet and MNASNet	UAV Camera
39	[73]	AlexNet, GoogLeNet, VGGNet and ResNet	Digital Camera
40	[74]	Faster R-CNN	UAV Camera
41	[75]	YOLOv5, RetinaNet, and Faster R-CNN	UAV Camera
42	[76]	Hybrid CNN, AlexNet, GoogLeNET, VGG-Net, ResNet, and GAN	Digital Camera
43	[77]	VGGNet, VGG16, VGG19 and SVM	Database
44	[78]	WeedFocusNet based on ResNet152v2	Database
45	[79]	Faster R-CNN and VGG16	Digital Camera
46	[80]	YOLOv7, YOLOv7-m, YOLOv7-x, YOLOv7-w6, YOLOv7-d6s, YOLOv5, YOLOv4 and Faster R-CNN	UAV Camera
47	[81]	Inception-V3, VGG-16 and ResNet-50	Digital Camera
48	[82]	YOLOv3, YOLOv3-tiny, YOLOv4 and YOLOv4-tiny	Digital Camera
49	[83]	DenseNet, EfficientNet and ResNet	Digital Camera
50	[84]	2D-CNN of specific design	UAV Camera
51	[85]	Alexnet, DarkNet53, GoogLeNet, InceptionV3, ResNet50 and Xception	Smartphone
52	[86]	Yolov8l, RetinaNet, and GAN	Digital Camera
53	[87]	AlexNet and CNN-RF species	UAV Camera
54	[88]	ResNet-18, YOLOv3, CenterNet, and Faster R-CNN	Smartphone
55	[89]	DenseHHO is based on Harris Hawk (HHO), DenseNet-121, and DenseNet-201 optimization algorithms.	UAV Camera
56	[90]	AlexNet and AlexNet -SVM	Database
57	[91]	YOLOv3, YOLOv3-tiny, YOLOv4 and YOLOv4-tiny	Digital Camera
58	[92]	CoFly-WeedDB is based on SegNet, VGG16, ResNet50, DenseNet121, EfficientNetB0 and MobileNetV2	UAV Camera

Table 1. Cont.

No.	Reference	CNN Architecture	Source of Image
59	[93]	Xception, VGG (16, 19), ResNet (50, 101, 152, 101v2, 152v2), InceptionV3, InceptionResNetV2, MobileNet, MobileNetV2, DenseNet (121, 169, 201), NASNetMobile, NASNetLarge	UAV Camera
60	[94]	MobileNetV3 and ShuffleNet	Smartphone
61	[95]	YOLOv3, Faster R-CNN, AlexNet, GoogLeNet and VGGNet	Digital Camera

Regarding the types of images found in the study, 56 articles used images in the RGB color space, captured with different types of cameras, four articles worked with multi-spectral images, captured with multi-spectral cameras integrated into UAVs, and one article used Generative Adversarial Networks (GAN) to create its own RGB images.

Figure 5 illustrates the frequency of use of the different CNNs for segmentation, detection, and classification of weeds. The YOLO family of algorithms with its multiple versions is the most commonly applied architecture, followed by the VGG, ResNet, and Faster R-CNN architectures, and the previous one in various versions. Alexnet and MobileNet are also commonly used.

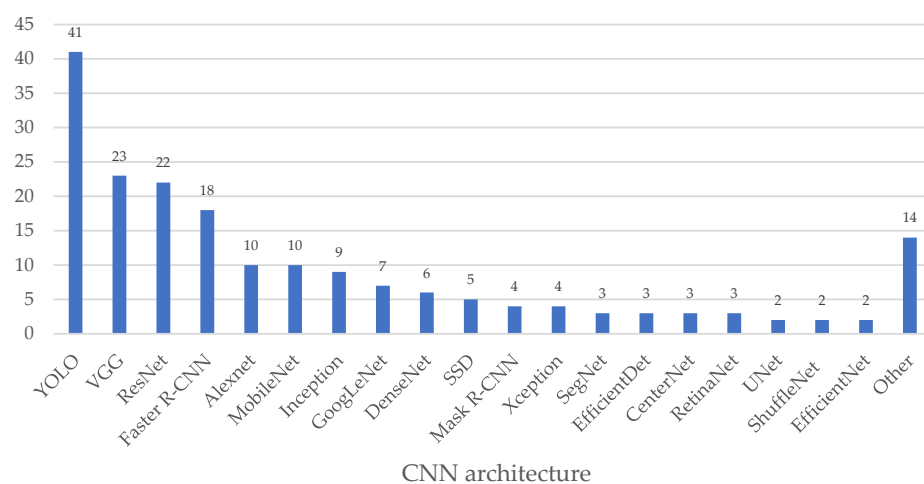


Figure 5. The frequency of use of the different CNN architectures found in this study.

As for the species used by the researchers, they are characterized by their great variety; a total of 27 species of cash crops and 77 species of weeds were counted. Figure 6 shows the frequency of the crop species used in the different studies, showing that the five most widely studied crops are Sugar Beet (*Beta vulgaris*), Soybean (*Glycine max*), Cotton (*Gossypium hirsutum*), Corn (*Zea mays*), and Wheat (*Triticum aestivum* L.).

Figure 7 shows the frequency of the families of the weed species used in the different studies. The weed species were grouped into their families due to the variety found, obtaining 22 in total. It was found that the most studied weed families are Poaceae, Asteraceae, Amaranthaceae, Convolvulaceae, and Brassicaceae.

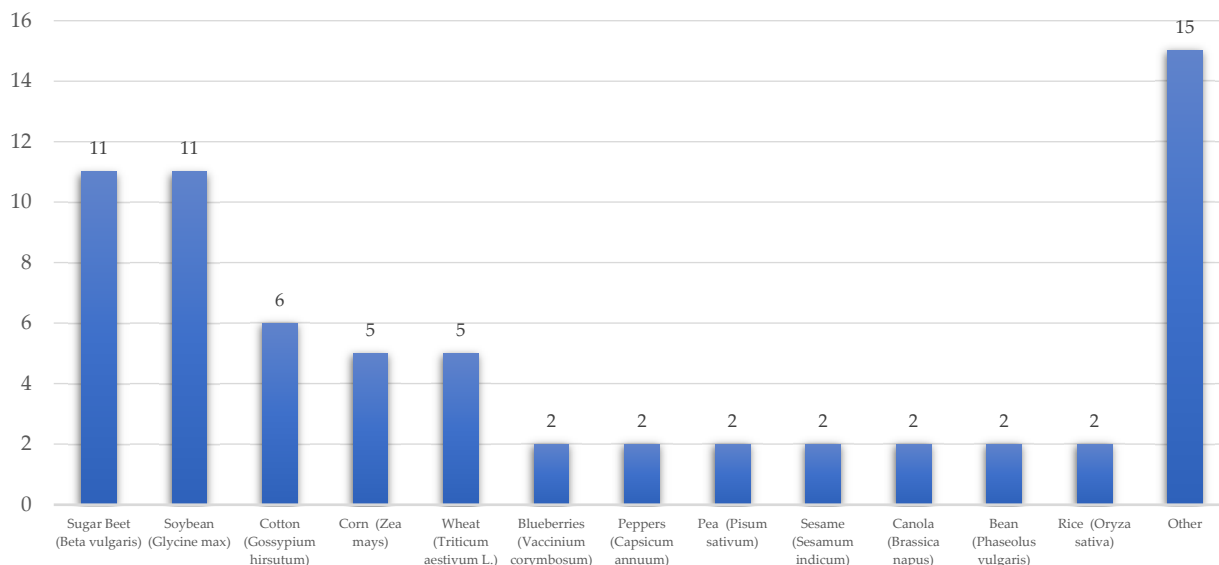


Figure 6. Frequency of crop species found in this study.

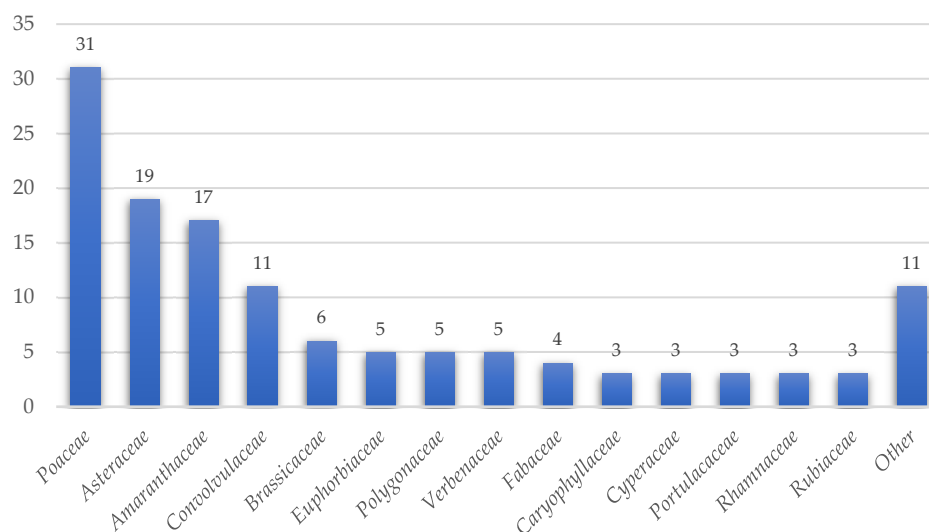


Figure 7. Frequency of weed families found in this study.

4. Discussion

Figure 5 shows that the YOLO family of algorithms with its different versions is used most frequently in the investigations analyzed, followed by the VGG family of algorithms. ResNet and Faster R-CNN are used less frequently. These architectures are noted for their high accuracy but differ in speed; the case of VGG, ResNet, and Faster R-CNN require more time to process images, which may not be practical for applications that need fast responses. On the other hand, the YOLO family of architectures is widely recognized for its efficiency in real-time object detection, with a high ability to predict bounding boxes and classify multiple objects in a single pass of the network, which makes it extremely fast and suitable for real-time detection applications, for example, [49,96] claim that YOLO CNNs have higher speed in detecting weeds.

The least frequently used architectures are UNet, ShuffleNet, and EfficientNet. In the case of UNet, it is mainly used in medical image segmentation applications, which explains its low frequency of use in weed detection, but different researchers have made comparisons seeking to evaluate its performance with other specific CNNs for object detection, for example, in [38], using the CNNs SegNet, UNet, VGG16, and ResNet-50, for

the detection of weeds in canola fields, the best model was SegNet. Similarly, the work of [63] combined U-Net with ResNet50, using it as a coding block for semantic segmentation of sugar beet, weeds, and soil, obtaining satisfactory results in the segmentation; this opens a window to evaluate different CNN architectures in different applications for which they were not designed. In the case of ShuffleNet and EfficientNet, these are architectures used in applications where computational efficiency is required; they are usually used in devices with limited resources, for tasks such as semantic segmentation, image classification, or resource optimization, which limits their direct applicability in object detection tasks and, in this case, for weed detection. Despite this, they have been used in works such as [72] which compared the CNNs VGG, Resnet, DenseNet, ShuffleNet, MobileNet, EfficientNet, and MNASNet to detect *Rumex obtusifolius* in grasslands and found that the best CNN was MobilNet.

Selecting the most suitable CNN architecture for weed detection should focus on aspects such as speed, accuracy, adaptability to different types of weeds and crops, environmental conditions, and computational efficiency. Although the YOLO family algorithms stand out for their real-time speed and ability to detect multiple objects in a single pass, other architectures, such as VGG, ResNet, and Faster R-CNN, offer higher accuracy. However, they may require more processing time. Therefore, speed becomes a parameter whose relevance depends on the application. For example, in developing in-field systems (weeding robots), where working speed is essential to ensure optimal performance, the fastest CNNs should be considered while maintaining a balance with accuracy. On the other hand, if precision is prioritized, hardware with higher processing capacity should be considered to increase speed as much as possible, leading to a higher economic cost in developing this type of machine.

In relation to the adaptability to different types of weeds and crops and various environmental conditions, the efficiency of CNNs is intrinsically related to the quality and diversity of the training set of images. This diversity should contain the highest representativeness of field conditions, ensuring the model can be effectively generalized to real conditions. Therefore, the training images must cover a wide range of scenarios and environmental conditions, including variations in weed types (morphological and color diversity), crops (variability in planting density), lighting conditions (brightness and shadows), soil textures, and crop residues, among other conditions present in the field. When selecting a CNN architecture for weed detection, carefully considering computational efficiency and available resources is essential. This involves balancing accuracy, efficiency, and speed to ensure optimal system performance.

One of the promises in weed detection and classification is the YOLO family of algorithms that stand out for their speed and efficiency in detecting objects in real-time. Other architectures, such as MobileNet, ShuffleNet, SqueezeNet, and EfficientNet, show promise in specific applications requiring particularly high computational efficiency or image segmentation with limited computational resources, such as mobile devices or development boards deployed in low-cost field systems. The constant evolution of algorithms allows any CNN to be implemented in field applications, e.g., VGG, ResNet, and Faster R-CNN offer higher accuracy but require high computational resources, which opens a window for researchers to optimize CNNs or combine them to reduce computational consumption and improve speed, while maintaining accuracy.

The selected papers analysed in this review show the use of a range of sources to capture images, from multi-spectral cameras on UAVs, to low-cost system development, to smartphones available to everyone. Several articles integrated different sources, as, for example, [60], who used a Nikon 7000 camera (Nikon Corporation, Tokyo, Japan) to build the image dataset for training using YOLOv5. Additionally, they built a spraying system using Raspberry cameras to distinguish dicotyledonous from monocotyledonous weeds. In [68] study, they used an UAV, a multi-copter drone (Hylio Inc., Houston, TX, USA) equipped with a Fujifilm GFX100 (100 MP) camera (Fujifilm Corporation, Tokyo, Japan), using YOLOv4 and Faster R-CNN architectures. In [37], they used two cameras, Sony

Cyber-Shot (Sony Corporation, Tokyo, Japan) and Canon EOS Rebel T6 (Canon Inc., Tokyo, Japan), for image acquisition, with AlexNet, GoogLeNet, and VGGNet architectures to detect Perennial ryegrass, dandelion (*Taraxacum officinale*), ground ivy (*Glechoma hederacea* L.) and spotted spurge (*Euphorbia maculata* L.).

The static digital camera is the most commonly used source of image capture due to its higher image quality and speed compared to that of a smartphone or a camera integrated into a UAV; in addition, the digital camera allows greater configuration of the capture adjustment parameters, unlike the cameras integrated into UAVs, which allow specific configurations but cannot be compared to the performance of professional cameras. Additionally, factors such as the movement of the UAV, either by wind or high speeds at the time of capture, cause some images to be out of focus, losing relevant information and making it difficult to label objects. On the other hand, the image quality depends on the type of camera sensor; the charge-coupled device (CCD) sensor is used in some digital cameras due to its image quality and lower noise, but they have a higher power consumption and manufacturing cost. The CCD has been replaced in many digital cameras by CMOS (Complementary metal-oxide-semiconductor) sensors that are more energy efficient and less expensive to manufacture but have a slightly lower quality than the CCD in low light conditions. These types of sensors are used in cameras integrated into UAVs and smartphones, so the choice between the two sensors depends on the application's needs. For example, [59] evaluated three digital cameras, a Canon T6 DSLR camera (Canon Inc., Tokyo, Japan), an LG G6 smartphone (LG Electronics, Seoul, South Korea), and a Logitech c920 camera (Logitech International S.A., Lausanne, Switzerland), and the detection results were higher for the Canon T6 camera. In contrast, the Logitech c920 camera was not suitable for weed detection, demonstrating that SLR-type cameras are preferred for the development of mobile platforms, field carts, or robots due to their image quality and different adjustment options. In the work of [65], a semi-professional Nikon P250 camera (Nikon Corporation, Tokyo, Japan) was used to develop a prototype autonomous sprayer; similarly, in [36], a field robot was designed to detect weeds in high-density crops using a 20-Mpixel JAI camera (JAI A/S, Copenhagen, Denmark), which is a high-speed industrial camera.

Regarding the cameras integrated into the UAVs, the cameras manufactured by DJI (DJI, Shenzhen, China) are mainly used in the Phantom 3 and 4, Matrice 600, Spark and Mavic versions. Therefore, they are limited to the performance and sensor configuration used by the brand. In addition, [72] used UAV Phantom 3 Professional imaging at three flight altitudes (10 m, 15 m, and 30 m); the VGG, ResNet, and DenseNet architectures, along with smaller ShuffleNet, MobileNet, EfficientNet, and MNASNet models were used to detect *Rumex obtusifolius*. In [51], they used a Mavic pro UAV integrated with a Parrot Sequoia camera, modified the CNN VGG-16 to detect weeds in sugar beet (*Beta vulgaris* subsp.), and used a Mavic pro UAV integrated with a Parrot Sequoia camera, modified the CNN VGG-16 to detect weeds in sugar beet (*Beta vulgaris* subsp.).

The use of smartphones for imaging source capture has increased in recent years, as there are more efficient and faster CNNs, such as MobileNet, specifically designed to be integrated into mobile devices that perform image capture and processing but lack professional settings. In the work of [62], a Huawei Y7 Prime smartphone (Huawei Technologies Co., Ltd., Shenzhen, China) was used to take images in pea (*Pisum sativum*) to work with the Faster RCNN ResNet 50 model. Similarly, [70], a Xiaomi Mi 11 smartphone (Xiaomi Corporation, Beijing, China) was used on bell pepper (*Capsicum annum* L.) to apply Alexnet, GoogLeNet, InceptionV3, and Xception. In addition, researchers take advantage of existing databases and image repositories, which are inexpensive, and many authors make images and training sets freely available. However, they do not exist for all weed species and crops. In [67], the DeepWeeds dataset was used to train SSD-MobileNet, SSD-InceptionV2, Faster R-CNN, CenterNet, EfficientDet, RetinaNet, and YOLOv4 models. Ref. [77] used the agri_data dataset, available on Kaggle, on Falsethistle grass and walnut (*Carya illinoensis*), to train VGGNet, VGG16, VGG19, and SVM models.

In terms of the infrastructure required to train CNNs, Graphics Processing Units (GPU) are necessary, allowing the processor to offload training work to take care of general management tasks, such as data loading, workflow coordination, and communication with other system components. NVIDIA, known primarily for its development and manufacturing of GPUs, developed a parallel computing platform, CUDA (Compute Unified Device Architecture), allowing developers to utilize NVIDIA GPUs' processing power. This technology has been instrumental in the field of artificial intelligence (AI), especially in the training and inference of CNNs, so many researchers use it to train and validate their architectures. The following are the versions of GPUs found in this study:

- Google Services—NVIDIA Tesla K80
- NVIDIA GeForce GTX Serie 10, versions 1050, 1050Ti, 1060, 1070 y 1080Ti.
- NVIDIA GeForce RTX Serie 20, versions 2060, 2070, 2070Ti, 2080, 2080Ti.
- NVIDIA GeForce RTX Serie 30, versión 3090.
- NVIDIA GeForce GTX TITAN X
- NVIDIA Quadro, versions RTX 5000, RTX 8000 y T2000
- NVIDIA Tesla A100
- NVIDIA TITAN X
- NVIDIA GeForce GT525M

5. Conclusions

In this systematic review, following the guidelines of the PRISMA statement, 61 scientific articles on the detection of weeds using CNN were analysed, using the following databases: Web of Science and Scopus. The review covers the last five years (2019–2023).

The CNN architectures were identified as the most commonly applied for weed segmentation, detection, and classification, followed by the YOLO family of algorithms, with its different versions. The VGG architecture and its versions are the third most widely used, followed by the ResNet and Faster R-CNN. The Alexnet and MobileNet were the least widely used architectures.

The sources used to acquire the images used to train and validate the CNNs were identified. Fixed digital cameras, reflex type, or low-cost cameras which allow a wide configuration and higher image quality are the main sources used. Cameras integrated into UAVs are also used, despite their speed limitation and lower configurations compared to an SLR camera, which can be RGB or multi-spectral. Smartphones with high resolution cameras have also been used, with the drawback of low processing speed. In this review, it was noted that some authors used free databases or databases from previous studies to avoid image acquisition and the difficulties associated with it.

Despite the demonstrated effectiveness of CNNs in weed detection, several limitations and challenges arise that deserve attention in future research. One critical limitation is the need for large data sets representing a wide range of weed species and images at different growth stages of both the weeds and the crop. Another significant disadvantage is the limited ability of CNNs to generalize new images under untrained environmental conditions. Therefore, datasets must include different lighting conditions, shadows, glare, and the presence of elements commonly found in the crop. This diversity of data is crucial to ensure that the model can effectively generalize across different scenarios encountered under field conditions.

However, collecting and labeling these datasets are costly and laborious tasks that often require experts, so using new techniques such as GANs is an alternative to exploring with TL. Likewise, it is necessary to explore combining CNN structures to make them less complex and shallower, implying less processing time.

Finally, CNNs require significant computational resources, especially when using deep architectures or large datasets. This computational demand can be a limitation in resource-constrained environments, such as mobile devices or embedded systems. Therefore, computational efficiency should be carefully considered when selecting a CNN architecture for applications in these environments, seeking a balance between model accuracy

and the required computational load. As future work, it is expected that the review will be extended to focus on the search for the most appropriate CNN architectures for weed detection and classification. Finally, it is hoped that this review article will help researchers to create new technological developments that will improve weed detection.

Author Contributions: Conceptualization, O.L.G.-N., A.C.-G. and L.M.N.-G.; methodology, O.L.G.-N., A.C.-G. and L.M.N.-G.; software, O.L.G.-N.; validation, O.L.G.-N. and L.M.N.-G.; formal analysis, O.L.G.-N., A.C.-G. and L.M.N.-G.; investigation, O.L.G.-N. and L.M.N.-G.; resources, L.M.N.-G.; writing—original draft preparation, O.L.G.-N. and L.M.N.-G.; writing—review and editing, O.L.G.-N., A.C.-G. and L.M.N.-G.; supervision, O.L.G.-N., A.C.-G. and L.M.N.-G.; project administration, A.C.-G. and L.M.N.-G.; funding acquisition, A.C.-G. and L.M.N.-G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the European Union through the Horizon Europe Program (HORIZON-CL6-2022-FARM2FORK-01) under project *Agro-ecological strategies for resilient farming in West Africa* (CIRAWA). Oscar Leonardo García-Navarrete has been financed under the call for University of Valladolid predoctoral contracts, co-financed by Banco Santander.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The authors thank Research Group TADRUS of the Department of Agricultural and Forestry Engineering, University of Valladolid.

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

References

1. *World Population Prospects 2022*; Department of Economic and Social Affairs Population Division. United Nations 2022. Available online: https://www.un.org/development/desa/pd/sites/www.un.org.development.desa.pd/files/wpp2022_summary_of_results.pdf (accessed on 12 October 2023).
2. Rajcan, I.; Swanton, C.J. Understanding Maize–Weed Competition: Resource Competition, Light Quality and the Whole Plant. *Field Crops Res.* **2001**, *71*, 139–150. [CrossRef]
3. Iqbal, N.; Manalil, S.; Chauhan, B.S.; Adkins, S.W. Investigation of Alternate Herbicides for Effective Weed Management in Glyphosate-Tolerant Cotton. *Arch. Agron. Soil. Sci.* **2019**, *65*, 1885–1899. [CrossRef]
4. Vii, M.; Williamson, M.; Lonsdale, M. Competition Experiments on Alien Weeds with Crops: Lessons for Measuring Plant Invasion Impact. *Biol. Invasions* **2004**, *6*, 59–69.
5. Holt, J.S. Principles of Weed Management in Agroecosystems and Wildlands. *Weed Technol.* **2004**, *18*, 1559–1562. [CrossRef]
6. Liu, W.; Xu, L.; Xing, J.; Shi, L.; Gao, Z.; Yuan, Q. Research Status of Mechanical Intra-Row Weed Control in Row Crops. *J. Agric. Mech. Res.* **2017**, *33*, 243–250.
7. Liu, B.; Bruch, R. Weed Detection for Selective Spraying: A Review. *Curr. Robot. Rep.* **2020**, *1*, 19–26. [CrossRef]
8. Hasan, A.S.M.M.; Soheli, F.; Diepeveen, D.; Laga, H.; Jones, M.G.K. A Survey of Deep Learning Techniques for Weed Detection from Images. *Comput. Electron. Agric.* **2021**, *184*, 106067. [CrossRef]
9. Al-Badri, A.H.; Ismail, N.A.; Al-Dulaimi, K.; Salman, G.A.; Khan, A.R.; Al-Sabaawi, A.; Salam, M.S.H. Classification of Weed Using Machine Learning Techniques: A Review—Challenges, Current and Future Potential Techniques. *J. Plant Dis. Prot.* **2022**, *129*, 745–768. [CrossRef]
10. Rai, N.; Zhang, Y.; Ram, B.G.; Schumacher, L.; Yellavajjala, R.K.; Bajwa, S.; Sun, X. Applications of Deep Learning in Precision Weed Management: A Review. *Comput. Electron. Agric.* **2023**, *206*, 107698. [CrossRef]
11. Mahmudul Hasan, A.S.M.; Soheli, F.; Diepeveen, D.; Laga, H.; Jones, M.G.K. Weed Recognition Using Deep Learning Techniques on Class-Imbalanced Imagery. *Crop Pasture Sci.* **2022**, *74*, 628–644. [CrossRef]
12. Radoglou-Grammatikis, P.; Sarigiannidis, P.; Lagkas, T.; Moscholios, I. A Compilation of UAV Applications for Precision Agriculture. *Comput. Netw.* **2020**, *172*, 107148. [CrossRef]
13. Chen, D.; Lu, Y.; Li, Z.; Young, S. Performance Evaluation of Deep Transfer Learning on Multi-Class Identification of Common Weed Species in Cotton Production Systems. *Comput. Electron. Agric.* **2022**, *198*, 107091. [CrossRef]
14. Adhinata, F.D.; Ramadhan, N.G.; Fauzi, M.D.; Ferani Tanjung, N.A. A Combination of Transfer Learning and Support Vector Machine for Robust Classification on Small Weed and Potato Datasets. *Int. J. Inform. Vis.* **2023**, *7*, 535–541. [CrossRef]
15. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A Comprehensive Survey on Transfer Learning. *Proc. IEEE* **2019**, *109*, 43–76. [CrossRef]
16. Wu, H.; Wang, Y.; Zhao, P.; Qian, M. Small-Target Weed-Detection Model Based on YOLO-V4 with Improved Backbone and Neck Structures. *Precis. Agric.* **2023**, *24*, 2149–2170. [CrossRef]

17. Olsen, A.; Konovalov, D.A.; Philippa, B.; Ridd, P.; Wood, J.C.; Johns, J.; Banks, W.; Girgenti, B.; Kenny, O.; Whinney, J.; et al. DeepWeeds: A Multiclass Weed Species Image Dataset for Deep Learning. *Sci. Rep.* **2019**, *9*, 2058. [[CrossRef](#)] [[PubMed](#)]
18. Suh, H.K.; IJsselmuiden, J.; Hofstee, J.W.; van Henten, E.J. Transfer Learning for the Classification of Sugar Beet and Volunteer Potato under Field Conditions. *Biosyst. Eng.* **2018**, *174*, 50–65. [[CrossRef](#)]
19. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012.
20. Espejo-Garcia, B.; Mylonas, N.; Athanasakos, L.; Fountas, S.; Vasilakoglou, I. Towards Weeds Identification Assistance through Transfer Learning. *Comput. Electron. Agric.* **2020**, *171*, 105306. [[CrossRef](#)]
21. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A Survey on Deep Transfer Learning. In Proceedings of the Artificial Neural Networks and Machine Learning—ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, 4–7 October 2018; Springer International Publishing: Berlin/Heidelberg, Germany, 2018.
22. Yan, X.; Deng, X.; Jin, J. Classification of Weed Species in the Paddy Field with DCNN-Learned Features. In Proceedings of the 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 12–14 June 2020; pp. 336–340.
23. Espejo-Garcia, B.; Mylonas, N.; Athanasakos, L.; Vali, E.; Fountas, S. Combining Generative Adversarial Networks and Agricultural Transfer Learning for Weeds Identification. *Biosyst. Eng.* **2021**, *204*, 79–89. [[CrossRef](#)]
24. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
25. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
26. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
27. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
28. Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
29. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
30. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
31. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; Kwon, Y.; Michael, K.; Fang, J.; Yifu, Z.; Wong, C.; Montes, D.; et al. Ultralytics/Yolov5: V7.0—YOLOv5 SOTA Realtime Instance Segmentation. *Zenodo* **2022**. [[CrossRef](#)]
32. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023.
33. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLO (Version 8.0.0). 2023. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 12 December 2023).
34. Urrútia, G.; Bonfill, X. PRISMA Declaration: A Proposal to Improve the Publication of Systematic Reviews and Meta-Analyses. *Med. Clin.* **2010**, *135*, 507–511. [[CrossRef](#)] [[PubMed](#)]
35. Elnemr, H.A. Convolutional Neural Network Architecture for Plant Seedling Classification. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*, 319–325. [[CrossRef](#)]
36. Rasti, P.; Ahmad, A.; Samiei, S.; Belin, E.; Rousseau, D. Supervised Image Classification by Scattering Transform with Application Toward Weed Detection in Culture Crops of High Density. *Remote Sens.* **2019**, *11*, 249. [[CrossRef](#)]
37. Yu, J.; Schumann, A.W.; Cao, Z.; Sharpe, S.M.; Boyd, N.S. Weed Detection in Perennial Ryegrass with Deep Learning Convolutional Neural Network. *Front. Plant Sci.* **2019**, *10*, 483304. [[CrossRef](#)] [[PubMed](#)]
38. Asad, M.H.; Bais, A. Weed Detection in Canola Fields Using Maximum Likelihood Classification and Deep Convolutional Neural Network. *Inf. Process. Agric.* **2020**, *7*, 535–545. [[CrossRef](#)]
39. Bah, M.D.; Hafiane, A.; Canals, R. CRowNet: Deep Network for Crop Row Detection in UAV Images. *IEEE Access* **2020**, *8*, 5189–5200. [[CrossRef](#)]
40. Gao, J.; French, A.P.; Pound, M.P.; He, Y.; Pridmore, T.P.; Pieters, J.G. Deep Convolutional Neural Networks for Image-Based *Convolvulus sepium* Detection in Sugar Beet Fields. *Plant Methods* **2020**, *16*, 29. [[CrossRef](#)]
41. Gupta, K.; Rani, R.; Bahia, N.K. Plant-Seedling Classification Using Transfer Learning-Based Deep Convolutional Neural Networks. *Int. J. Agric. Environ. Inf. Syst.* **2020**, *11*, 25–40. [[CrossRef](#)]
42. Mora-Fallas, A.; Goeau, H.; Joly, A.; Bonnet, P.; Mata-Montero, E. Instance Segmentation for Automated Weeds and Crops Detection in Farmlands. *Tecnol. Marcha* **2020**, *33*, 13–17. [[CrossRef](#)]
43. Osorio, K.; Puerto, A.; Pedraza, C.; Jamaica, D.; Rodríguez, L. A Deep Learning Approach for Weed Detection in Lettuce Crops Using Multispectral Images. *AgriEngineering* **2020**, *2*, 471–488. [[CrossRef](#)]
44. Parico, A.I.B.; Ahamed, T. An Aerial Weed Detection System for Green Onion Crops Using the You Only Look Once (YOLOv3) Deep Learning Algorithm. *Eng. Agric. Environ. Food* **2020**, *13*, 42–48. [[CrossRef](#)] [[PubMed](#)]

45. Sivakumar, A.N.V.; Li, J.; Scott, S.; Psota, E.; Jhala, A.J.; Luck, J.D.; Shi, Y. Comparison of Object Detection and Patch-Based Classification Deep Learning Models on Mid-to Late-Season Weed Detection in UAV Imagery. *Remote Sens.* **2020**, *12*, 2136. [[CrossRef](#)]
46. Haq, M.A. CNN Based Automated Weed Detection System Using UAV Imagery. *Comput. Syst. Sci. Eng.* **2021**, *42*, 837–849. [[CrossRef](#)]
47. Hennessy, P.J.; Esau, T.J.; Farooque, A.A.; Schumann, A.W.; Zaman, Q.U.; Corscadden, K.W. Hair Fescue and Sheep Sorrel Identification Using Deep Learning in Wild Blueberry Production. *Remote Sens.* **2021**, *13*, 943. [[CrossRef](#)]
48. Hu, C.; Thomasson, J.A.; Bagavathiannan, M.V. A Powerful Image Synthesis and Semi-Supervised Learning Pipeline for Site-Specific Weed Detection. *Comput. Electron. Agric.* **2021**, *190*, 106423. [[CrossRef](#)]
49. Jabir, B.; Falih, N.; Rahmani, K. Accuracy and Efficiency Comparison of Object Detection Open-Source Models. *Int. J. Online Biomed. Eng.* **2021**, *17*, 165–184. [[CrossRef](#)]
50. Khan, S.; Tufail, M.; Khan, M.T.; Khan, Z.A.; Anwar, S. Deep Learning-Based Identification System of Weeds and Crops in Strawberry and Pea Fields for a Precision Agriculture Sprayer. *Precis. Agric.* **2021**, *22*, 1711–1727. [[CrossRef](#)]
51. Moazzam, S.I.; Khan, U.S.; Qureshi, W.S.; Tiwana, M.I.; Rashid, N.; Alasmay, W.S.; Iqbal, J.; Hamza, A. A Patch-Image Based Classification Approach for Detection of Weeds in Sugar Beet Crop. *IEEE Access* **2021**, *9*, 121698–121715. [[CrossRef](#)]
52. Urmashv, B.; Buribayev, Z.; Amirgaliyeva, Z.; Ataniyazova, A.; Zhassuzak, M.; Turegali, A. Development of a Weed Detection System Using Machine Learning and Neural Network Algorithms. *East.-Eur. J. Enterp. Technol.* **2021**, *6*, 70–85. [[CrossRef](#)]
53. Xie, S.; Hu, C.; Bagavathiannan, M.; Song, D. Toward Robotic Weed Control: Detection of Nutsedge Weed in Bermudagrass Turf Using Inaccurate and Insufficient Training Data. *IEEE Robot. Autom. Lett.* **2021**, *6*, 7365–7372. [[CrossRef](#)]
54. Xu, K.; Zhu, Y.; Cao, W.; Jiang, X.; Jiang, Z.; Li, S.; Ni, J. Multi-Modal Deep Learning for Weeds Detection in Wheat Field Based on RGB-D Images. *Front. Plant Sci.* **2021**, *12*, 732968. [[CrossRef](#)] [[PubMed](#)]
55. Al-Badri, A.H.; Ismail, N.A.; Al-Dulaimi, K.; Rehman, A.; Abunadi, I.; Bahaj, S.A. Hybrid CNN Model for Classification of Rumex Obtusifolius in Grassland. *IEEE Access* **2022**, *10*, 90940–90957. [[CrossRef](#)]
56. Babu, V.S.; Ram, N.V. Deep Residual CNN with Contrast Limited Adaptive Histogram Equalization for Weed Detection in Soybean Crops. *Trait. Signal* **2022**, *39*, 717–722. [[CrossRef](#)]
57. Chen, J.; Wang, H.; Zhang, H.; Luo, T.; Wei, D.; Long, T.; Wang, Z. Weed Detection in Sesame Fields Using a YOLO Model with an Enhanced Attention Mechanism and Feature Fusion. *Comput. Electron. Agric.* **2022**, *202*, 107412. [[CrossRef](#)]
58. Sunil, G.C.; Koparan, C.; Ahmed, M.R.; Zhang, Y.; Howatt, K.; Sun, X. A Study on Deep Learning Algorithm Performance on Weed and Crop Species Identification under Different Image Background. *Artif. Intell. Agric.* **2022**, *6*, 242–256. [[CrossRef](#)]
59. Hennessy, P.J.; Esau, T.J.; Schumann, A.W.; Zaman, Q.U.; Corscadden, K.W.; Farooque, A.A. Evaluation of Cameras and Image Distance for CNN-Based Weed Detection in Wild Blueberry. *Smart Agric. Technol.* **2022**, *2*, 100030. [[CrossRef](#)]
60. Jabir, B.; Falih, N. Deep Learning-Based Decision Support System for Weeds Detection in Wheat Fields. *Int. J. Electr. Comput. Eng.* **2022**, *12*, 816–825. [[CrossRef](#)]
61. Liu, S.; Jin, Y.; Ruan, Z.; Ma, Z.; Gao, R.; Su, Z. Real-Time Detection of Seedling Maize Weeds in Sustainable Agriculture. *Sustainability* **2022**, *14*, 15088. [[CrossRef](#)]
62. Mohammed, H.; Tannouche, A.; Ounejar, Y. Weed Detection in Pea Cultivation with the Faster RCNN ResNet 50 Convolutional Neural Network. *Rev. D'intelligence Artif.* **2022**, *36*, 13–18. [[CrossRef](#)]
63. Nasiri, A.; Omid, M.; Taheri-Garavand, A.; Jafari, A. Deep Learning-Based Precision Agriculture through Weed Recognition in Sugar Beet Fields. *Sustain. Comput. Inform. Syst.* **2022**, *35*, 100759. [[CrossRef](#)]
64. Razfar, N.; True, J.; Bassiouny, R.; Venkatesh, V.; Kashef, R. Weed Detection in Soybean Crops Using Custom Lightweight Deep Learning Models. *J. Agric. Food Res.* **2022**, *8*, 100308. [[CrossRef](#)]
65. Saboia, H.d.S.; Mion, R.L.; Silveira, A.D.O.; Mamiya, A.A. Real-time selective spraying for viola rope control in soybean and cotton crops using deep learning. *Eng. Agric.* **2022**, *42*, e20210163. [[CrossRef](#)]
66. Saleem, M.H.; Potgieter, J.; Arif, K.M. Weed Detection by Faster RCNN Model: An Enhanced Anchor Box Approach. *Agronomy* **2022**, *12*, 1580. [[CrossRef](#)]
67. Saleem, M.H.; Velayudhan, K.K.; Potgieter, J.; Arif, K.M. Weed Identification by Single-Stage and Two-Stage Neural Networks: A Study on the Impact of Image Resizers and Weights Optimization Algorithms. *Front. Plant Sci.* **2022**, *13*, 850666. [[CrossRef](#)] [[PubMed](#)]
68. Sapkota, B.B.; Hu, C.; Bagavathiannan, M.V. Evaluating Cross-Applicability of Weed Detection Models Across Different Crops in Similar Production Environments. *Front. Plant Sci.* **2022**, *13*, 837726. [[CrossRef](#)] [[PubMed](#)]
69. Sapkota, B.B.; Popescu, S.; Rajan, N.; Leon, R.G.; Reberg-Horton, C.; Mirsky, S.; Bagavathiannan, M.V. Use of Synthetic Images for Training a Deep Learning Model for Weed Detection and Biomass Estimation in Cotton. *Sci. Rep.* **2022**, *12*, 19580. [[CrossRef](#)]
70. Subeesh, A.; Bhole, S.; Singh, K.; Chandel, N.S.; Rajwade, Y.A.; Rao, K.V.R.; Kumar, S.P.; Jat, D. Deep Convolutional Neural Network Models for Weed Detection in Polyhouse Grown Bell Peppers. *Artif. Intell. Agric.* **2022**, *6*, 47–54. [[CrossRef](#)]
71. Tannouche, A.; Gaga, A.; Boutalline, M.; Belhouideg, S. Weeds Detection Efficiency through Different Convolutional Neural Networks Technology. *Int. J. Electr. Comput. Eng.* **2022**, *12*, 1048–1055. [[CrossRef](#)]
72. Valente, J.; Hiremath, S.; Ariza-Sentís, M.; Doldersum, M.; Kooistra, L. Mapping of Rumex Obtusifolius in Nature Conservation Areas Using Very High Resolution UAV Imagery and Deep Learning. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *112*, 102864. [[CrossRef](#)]

73. Yang, J.; Bagavathiannan, M.; Wang, Y.; Chen, Y.; Yu, J. A Comparative Evaluation of Convolutional Neural Networks, Training Image Sizes, and Deep Learning Optimizers for Weed Detection in Alfalfa. *Weed Technol.* **2022**, *36*, 512–522. [[CrossRef](#)]
74. Ajayi, O.G.; Ashi, J. Effect of Varying Training Epochs of a Faster Region-Based Convolutional Neural Network on the Accuracy of an Automatic Weed Classification Scheme. *Smart Agric. Technol.* **2023**, *3*, 100128. [[CrossRef](#)]
75. Almalky, A.M.; Ahmed, K.R. Deep Learning for Detecting and Classifying the Growth Stages of *Consolida Regalis* Weeds on Fields. *Agronomy* **2023**, *13*, 934. [[CrossRef](#)]
76. Arif, S.; Hussain, R.; Ansari, N.M.; Rauf, W. A Novel Hybrid Feature Method for Weeds Identification in the Agriculture Sector. *Res. Agric. Eng.* **2023**, *69*, 132–142. [[CrossRef](#)]
77. Bidve, V.; Mane, S.; Tamkhade, P.; Pakle, G. Weed Detection by Using Image Processing. *Indones. J. Electr. Eng. Comput. Sci.* **2023**, *30*, 341–349. [[CrossRef](#)]
78. Devi, B.S.; Sandhya, N.; Chatrapati, K.S. WeedFocusNet: A Revolutionary Approach Using the Attention-Driven ResNet152V2 Transfer Learning. *Int. J. Recent Innov. Trends Comput. Commun.* **2023**, *11*, 334–341. [[CrossRef](#)]
79. Fan, X.; Chai, X.; Zhou, J.; Sun, T. Deep Learning Based Weed Detection and Target Spraying Robot System at Seedling Stage of Cotton Field. *Comput. Electron. Agric.* **2023**, *214*, 108317. [[CrossRef](#)]
80. Gallo, I.; Rehman, A.U.; Dehkordi, R.H.; Landro, N.; La Grassa, R.; Boschetti, M. Deep Object Detection of Crop Weeds: Performance of YOLOv7 on a Real Case Dataset from UAV Images. *Remote Sens.* **2023**, *15*, 539. [[CrossRef](#)]
81. Husham Al-Badri, A.; Azman Ismail, N.; Al-Dulaimi, K.; Ahmed Salman, G.; Sah Hj Salam, M. Adaptive Non-Maximum Suppression for Improving Performance of Rumex Detection. *Expert. Syst. Appl.* **2023**, *219*, 119634. [[CrossRef](#)]
82. Jannah, L.L.; Zhang, Y.; Cui, Z.; Yang, Y. Multi-Level Feature Re-Weighted Fusion for the Semantic Segmentation of Crops and Weeds. *J. King Saud Univ. Comput. Inf. Sci.* **2023**, *35*, 101545. [[CrossRef](#)]
83. Jin, X.; Liu, T.; McCullough, P.E.; Chen, Y.; Yu, J. Evaluation of Convolutional Neural Networks for Herbicide Susceptibility-Based Weed Detection in Turf. *Front. Plant Sci.* **2023**, *14*, 1096802. [[CrossRef](#)]
84. Kansal, I.; Khullar, V.; Verma, J.; Popli, R.; Kumar, R. IoT-Fog-Enabled Robotics-Based Robust Classification of Hazy and Normal Season Agricultural Images for Weed Detection. *Paladyn* **2023**, *14*, 20220105. [[CrossRef](#)]
85. Modi, R.U.; Kancheti, M.; Subeesh, A.; Raj, C.; Singh, A.K.; Chandel, N.S.; Dhimate, A.S.; Singh, M.K.; Singh, S. An Automated Weed Identification Framework for Sugarcane Crop: A Deep Learning Approach. *Crop Prot.* **2023**, *173*, 106360. [[CrossRef](#)]
86. Moreno, H.; Gómez, A.; Altares-López, S.; Ribeiro, A.; Andújar, D. Analysis of Stable Diffusion-Derived Fake Weeds Performance for Training Convolutional Neural Networks. *Comput. Electron. Agric.* **2023**, *214*, 108324. [[CrossRef](#)]
87. Ong, P.; Teo, K.S.; Sia, C.K. UAV-Based Weed Detection in Chinese Cabbage Using Deep Learning. *Smart Agric. Technol.* **2023**, *4*, 100181. [[CrossRef](#)]
88. Patel, J.; Ruparelia, A.; Tanwar, S.; Alqahtani, F.; Tolba, A.; Sharma, R.; Raboaca, M.S.; Neagu, B.C. Deep Learning-Based Model for Detection of Brinjal Weed in the Era of Precision Agriculture. *Comput. Mater. Contin.* **2023**, *77*, 1281–1301. [[CrossRef](#)]
89. Rajeena, P.P.F.; Ismail, W.N.; Ali, M.A.S. A Metaheuristic Harris Hawks Optimization Algorithm for Weed Detection Using Drone Images. *Appl. Sci.* **2023**, *13*, 7083. [[CrossRef](#)]
90. Reddy, B.S.; Neeraja, S. An Optimal Superpixel Segmentation Based Transfer Learning Using AlexNet–SVM Model for Weed Detection. *Int. J. Syst. Assur. Eng. Manag.* **2023**. [[CrossRef](#)]
91. Saqib, M.A.; Aqib, M.; Tahir, M.N.; Hafeez, Y. Towards Deep Learning Based Smart Farming for Intelligent Weeds Management in Crops. *Front. Plant Sci.* **2023**, *14*, 1211235. [[CrossRef](#)]
92. Shahi, T.B.; Dahal, S.; Sitaula, C.; Neupane, A.; Guo, W. Deep Learning-Based Weed Detection Using UAV Images: A Comparative Study. *Drones* **2023**, *7*, 624. [[CrossRef](#)]
93. Singh, V.; Gourisaria, M.K.; Harshvardhan, G.M.; Choudhury, T. Weed Detection in Soybean Crop Using Deep Neural Network. *Pertanika J. Sci. Technol.* **2023**, *31*, 401–423. [[CrossRef](#)]
94. Yu, H.; Che, M.; Yu, H.; Ma, Y. Research on Weed Identification in Soybean Fields Based on the Lightweight Segmentation Model DCSAnet. *Front. Plant Sci.* **2023**, *14*, 1268218. [[CrossRef](#)] [[PubMed](#)]
95. Zhuang, J.; Jin, X.; Chen, Y.; Meng, W.; Wang, Y.; Yu, J.; Muthukumar, B. Drought Stress Impact on the Performance of Deep Convolutional Neural Networks for Weed Detection in Bahiagrass. *Grass Forage Sci.* **2023**, *78*, 214–223. [[CrossRef](#)]
96. García-Navarrete, O.L.; Santamaria, O.; Martín-Ramos, P.; Valenzuela-Mahecha, M.Á.; Navas-Gracia, L.M. Development of a Detection System for Types of Weeds in Maize (*Zea mays* L.) under Greenhouse Conditions Using the YOLOv5 v7.0 Model. *Agriculture* **2024**, *14*, 286. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.