sensors

MDPI

Article

Effect of Depth Band Replacement on Red, Green and Blue Image for Deep Learning Weed Detection

Jan Vandrol, Janis Perren and Adrian Koller *

Institute of Mechanical Engineering and Energy Technology, Lucerne University of Applied Sciences and Arts, CH-6048 Horw, Switzerland
* Correspondence: adrian.koller@hslu.ch; Tel.: +41-41-349-31-03

Abstract: Automated agricultural robots are becoming more common with the decreased cost of sensor devices and increased computational capabilities of single-board computers. Weeding is one of the mundane and repetitive tasks that robots could be used to perform. The detection of weeds in crops is now common, and commercial solutions are entering the market rapidly. However, less work is carried out on combatting weeds in pastures. Weeds decrease the grazing yield of pastures and spread over time. Mowing the remaining weeds after grazing is not guaranteed to remove entrenched weeds. Periodic but selective cutting of weeds can be a solution to this problem. However, many weeds share similar textures and structures with grazing plants, making their detection difficult using the classic RGB (Red, Green, Blue) approach. Pixel depth estimation is considered a viable source of data for weed detection. However, systems utilizing RGBD (RGB plus Depth) are computationally expensive, making them nonviable for small, lightweight robots. Substituting one of the RGB bands with depth data could be a solution to this problem. In this study, we examined the effect of band substitution on the performance of lightweight YOLOv8 models using precision, recall and mAP50 metrics. Overall, the RDB band combination proved to be the best option for YOLOv8 small and medium detection models, with 0.621 and 0.634 mAP50 (for a mean average precision at 50% intersection over union) scores, respectively. In both instances, the classic RGB approach yielded lower accuracies of 0.574 and 0.613.

Keywords: RGBD; YOLO8; object detection; weed detection; deep learning; band substitution

check for updates

Academic Editor: Yongwha Chung

Received: 2 December 2024
Revised: 18 December 2024
Accepted: 18 December 2024
Published: 30 December 2024

Citation: Vandrol, J.; Perren, J.; Koller, A. Effect of Depth Band Replacement on Red, Green and Blue Image for Deep Learning Weed Detection. *Sensors* **2025**, *25*, 161. https://doi.org/10.3390/s25010161

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

Many grazing pastures suffer from weed issues. The low value creation on these fields prohibits expensive weed management activities, which leads to pastures that are often untended, with progressively larger outgrowths of plants which farm animals do not eat or are of a low nutritional value. Furthermore, invasive species may become more easily entrenched. Even when the part of the plant that is above the ground is cut, the roots can be strong enough to continue growing. Mowing the whole pasture after grazing may also be suboptimal, because over time, this favors vegetatively over generatively procreating plants. While this may increase the nutritional value (protein and total digestible nutrients), it will reduce the fiber and structure of the forage, which are necessary for a healthy animal gut [1].

The periodic defoliation of broad-leaf dock—a particularly persistent weed—was shown to weaken the plants enough to keep them in check [2]. Robot-based defoliation of these weeds would require a lightweight, computationally affordable detection solution that can be fitted onto compact rovers and, with high accuracy, distinguish between grazing plants and weeds.

The automated discrimination between weeds and grazing plants is challenging, as they often share similar textures and structures. Several studies [3–5] outlined the following difficulties with weed detection: occlusion, similarity in color and texture, plants being shadowed in natural light, color and texture variations due to lighting conditions, similarities in the structures of different species and changes in plant structure during different growth phases. Originally, weed detection was solved in multiple stages—preprocessing, vegetation extraction, feature extraction and classification [4]. However, advancements in both hardware and software solutions have given rise to the employment of deep learning (DL) techniques in weed detection [5,6].

DL is a subdiscipline of Machine Learning (ML) that uses multiple layers of neural networks to extract increasingly more abstract information from input data. In the context of image processing, this can mean moving from object edges to object structures. Thanks to this, DL approaches have an advantage over previously used methods. The similarity between plants can be overcome by the robust learning capabilities of DL. Agriculturally focused studies do indeed report better performance of DL algorithms over traditional approaches [4,7].

With the superior capabilities of DL methods in terms of recognizing structure and texture, some of the issues in weed detection can be addressed. There are, however, still issues of variable lighting conditions. To alleviate this, another source of information apart from the usual RGB data could be used: depth. Depth data—the distance of objects from a sensor—can be obtained using various means. Sensors using structured light, time of flight or vision methods exist. In the context of this paper, vision is the most logical choice, since RGB data need to be collected. While it is possible to obtain depth estimates using monocular cameras [8], a stereo camera is a more robust option. Stereo cameras have become smaller and cheaper in recent years, to the point where there are only a few advantages to using monocular cameras in autonomous tasks. Depth has several beneficial attributes for weed detection. It is fully invariant to illumination and color changes [9]. It also provides additional information about plant structures. Additionally, the weeds in the pastures will be more prominent, as they are not grazed on.

Considering the potential advantage of including depth for weed detection, it would be optimal to include it in the detection process. Unfortunately, the computational requirements for utilizing RGBD information in object detection are higher compared to regular RGB methods. For example, Takahashi et al. [10] reported a 40% decrease in frames per second (from 74 to 44 on a dedicated workstation using GTX 1080Ti) for their RGBD processing method. This leads to significantly higher power requirements, which is detrimental for battery-powered autonomous vehicles.

While RGB bands and depth convey different kinds of information, replacing one of the colors with depth data might yield additional information while having a low impact on the texture patterns. The question then is whether substitution of one of the RGB bands with depth information will lead to a better prediction accuracy while maintaining a lower computational cost. Our paper aims to answer this question with a comparative analysis of object detection results, utilizing different band combinations.

The paper is organized using the following structure: Section 2 contains a review of RGB and RGBD object detection methods and relevant weed detection efforts. Section 3 details the method used to evaluate RGB band substitution in weed detection, with the results being outlined in Section 4. The paper concludes with the discussion in Section 5.

2. Related Work

2.1. Deep Learning for Weed Detection

Over the years, many methods have been developed for recognizing weeds, be it color transformation [11], segmentation [12], plant morphology [13] or spatial distribution (plant row identification) [14]. DL in weed detection became popular for its ease of use and robustness [7].

Hasan et al. [15] tested several DL architectures on four weed types in corn fields. The DL methods were YOLOv7, YOLOv8 and Faster-RCNN (Region-based Convolutional Neural Networks). The dataset contained 5997 images with augmentations. The results showed that both YOLOv7 and YOLOv8 are capable of high mean average precision values: 89.93% and 89.39%, respectively. Such good performance allows for the use of automated field robots and drones in precision agriculture.

One such example is an object detection system for precision sprayers for strawberry and pea fields, developed by Khan et al. [16]. They collected 2800 images of weeds which were downsampled to a resolution of 448 × 448 pixels and input into a customized detection framework. The framework was based on a Faster-RCNN architecture with a ResNet-101 network and tailored anchor points. The resulting model achieved 95.3% weed detection accuracy.

Another example is the work by Asad and Bais [17], who proposed a weed mapping method with a semantic segmentation step during the labeling process. The segmentation of plants from the background was achieved using maximum likelihood classification. Pixel-level data instead of bounding boxes were then presented to a SegNet DL architecture with a ResNet-50 extractor. The resulting model was able to localize weeds with 82% accuracy.

In the context of weed detection, depth has also been used to improve results. Xu et al. [18] used RGBD data for detecting weeds in wheat fields. The depth image was converted to a PHA (phase, height and angle) image to convey additional information and have the same structure as RGB images. The framework used in this study comprised three detection networks—two specific ones to each image and one shared. The networks were based on Faster R-CNN and a VGG16 architecture. The three outputs were fused using individual weights at the decision level of the algorithm. The study reported a 42.9% mean average precision when detecting broad-leaf weeds.

2.2. RGBD in Object Detection

Depth was found to be beneficial in object detection tasks, even in cases where the depth has only been estimated from RGB images [19]. Hou et al. [9] argued that as depth carries different types of information from RGB, it should not be used at the same input level but have its own network for processing. They proposed calculating the horizontal disparity and pixel angle from the depth and combining them into a three-band image to be used alongside RGB.

Unprocessed depth information has also been used in object detection. Chu et al. [20] developed an outdoor surveillance system comprising two sub-networks, one for RGB and one for depth. The outputs of these networks were fused into a cross-modal object feature set. This set was then used in their depth-adaptive anchor generation module (DAGM). Anchors are bounding boxes of objects and are generated during training based on the object scale and aspect ratio. However, identical objects appear smaller when viewed from a distance. This can cause a decrease in accuracy when working with variable depths in images. The use of the DAGM and depth improved the accuracy of object detectors by approximately 2%.

3. Materials and Methods

3.1. Weed Selection

Broad-leaved dock (*Rumex obtusifolius*) was chosen as the target of this study. It is a highly invasive perennial weed and common in all temperate climate zones [21]. It is capable of withstanding extreme conditions (both drought and cold). Its robust and aggressive root system makes it difficult to eradicate by single mowing, and grazing animals avoid it. It can therefore quickly spread through grassland and decrease its feeding value. These attributes make it a prime target for automated repeated cutting efforts.

3.2. Image Acquisition

As of the time of writing, there were no public image datasets of broad-leaved dock containing depth information. A dataset of 588 images was gathered using a ZED2i stereo camera developed by Stereolabs. This camera is capable of capturing stereo images in multiple resolutions, ranging from 2208 × 1242 to 662 × 376. It has a baseline of 12 cm and a focal length of 4 mm, providing a depth range of up to 35 m. Its reported depth error is less than 2 percent for up to a 10 m depth [22]. The field of view is 72° × 44°. The camera was mounted at a constant height of 190 cm above the ground and pointed downwards. This gave us an image area of approximately 2.8 × 1.5 m. The camera was set to a 2208 × 1242 resolution during data collection.

Images of the vegetative, budding and flowering growth stages of the broad-leaved dock were taken to have a representation of different plant structures. Additionally, 10% of the dataset does not contain any weed images to alleviate bias during learning.

3.3. Image Preprocessing

First, the depth estimates and RGB images had to be combined. Depth data that are aligned with left or right stereo images can be accessed using Stereolabs SDK. However, the depth map did not cover a whole image area, as it was calculated from both left and right images. Therefore, each image width was reduced by 300 pixels to remove an area where no depth data could be estimated. Then, normalized depth data were added to the image as a fourth band. The images were also resized to a square for use in object detection training. A resolution of 1280 × 1280 was used as a base dataset. It can be accessed online [23]. A resolution of 640 × 640 was used for the actual deep learning training to facilitate inference on lightweight single-board computers. The aspect ratio was maintained by adding black borders on the top and bottom parts of the image.

The images were labeled using Label Studio software (ver. 1.9.0) [24]. A bounding box approach to labeling using YOLO formatting (center-point X and Y coordinates, width and height) was used. Image augmentation was also applied to decrease the chances of detection model overfitting and improve the overall performance, as described in other studies [25,26]. Noise generation and image transformation augmentation were introduced into the dataset, increasing the number of images to 1176. The images were randomly resized and rotated. Afterwards, Gaussian noise was also added. An albumentation library [27] was used for the task. The dataset was divided into training, validation and test sets with a ratio of 7:2:1. A data loader script was created to use different combinations of RGBD bands, virtually creating four different image datasets for object detection training. Figure 1 provides an example of the various combinations.

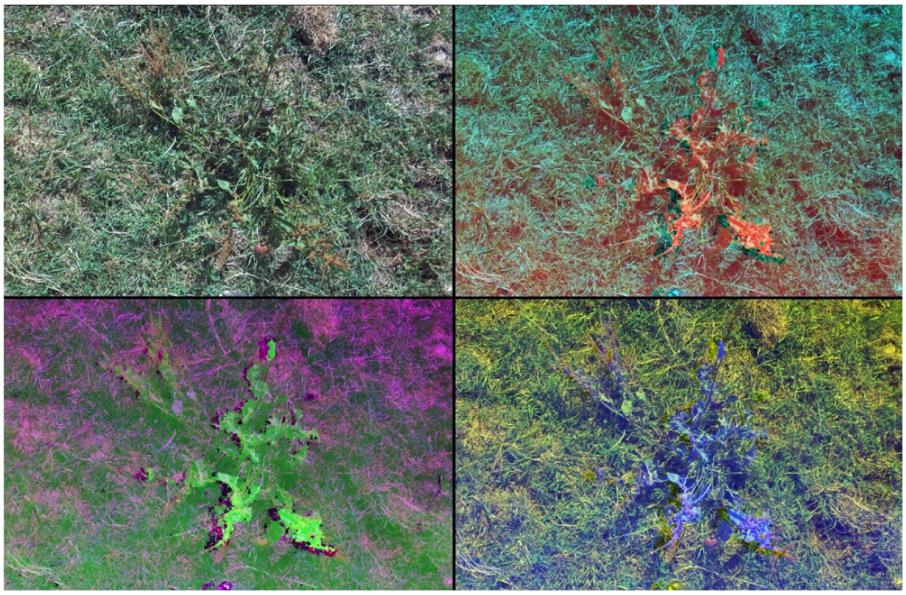


Figure 1. Result of different substitutions of RGB channels with depth. Combinations are as follows: top left—RGB; top right—DGB; bottom left—RDB; bottom right—RGD.

3.4. Object Detection Training

For testing of the effects of depth band substitution, the YOLO (You Only Look Once) detection model was selected. YOLO models have been found to be highly effective in weed detection in other studies [28–32]. YOLO was first developed in 2015 [33] and became the state of the art in many fields. It introduced a shift in the way that objects were detected from a two-shot to a single-shot system. Two-shot detectors (Faster RCNN or Region-based Fully Convolutional Networks, etc.) first predict the locations of objects in the image and then attempt to classify them. Single-shot detectors (SSD, YOLO) do everything in one pass through the neural network. While this approach loses in terms of accuracy, the increase in speed is substantial and allows for the use of real-time detection on a wide array of devices.

YOLO uses a single CNN for detection. It divides the image into a grid. For each grid cell, the bounding boxes and class confidence are calculated simultaneously. Then, a non-maximum suppression algorithm removes all but the strongest candidates, thus eliminating redundant or erroneous detections.

YOLO was released as an open-source project. Its architecture and algorithms have been iterated multiple times, creating a range of versions. At the time of writing, YOLOv9 has been released. However, in this project, YOLOv8 [34] was used. It comes with various improvements in architecture, anchor-free detections, new convolutions and mosaic augmentations. Overall, it shows an increase in performance (both speed and accuracy) when compared to its predecessors. YOLOv8 provides different model versions with increased complexity:

- YOLOv8n—3.2 million parameters;
- YOLOv8s—11.2 million parameters;
- YOLOv8m—25.9 million parameters;
- YOLOv8l—43.7 million parameters;
- YOLOv8x—68.2 million parameters.

Model parameters are directly tied to the computational complexity of its training and inference, as well as its potential to learn patterns, resulting in an accuracy increase. As the larger models are too complex to be run on single-board computers, only v8n, v8s and v8m models were considered. Models were run from scratch, and no pre-trained weights were used. There is a possibility that pre-trained models which did not include plants would

decrease the accuracy [35]. The substitution of depth for one of the color bands could also cause issues for the pre-trained model.

To provide comparable results, the hyperparameters were kept constant for all training runs. If changes to hyperparameters were made between runs, it would become challenging to determine the effect of the band substitution on the model's performance. Table 1 shows the values which were chosen, based on the AdamW hyperparameter optimizer. After training, the results of the model runs were also converted into an ONNX format (Open Neural Network Exchange) to improve the inference speed for RPi5 CPU-only processing.

Table 1. Hyperparameter values used in model training.

Hyperparameter	Value
Initial learning rate	0.002
Final learning rate	0.1
Momentum	0.9
Weight decay	0.0005
Training epochs	300
Batch size	32

3.5. Evaluation Metrics

The following metrics were chosen to assess the performance of resulting models, based on object detection studies [15,25,28,36]:

- Precision;
- Recall;
- mAP50;
- Inference speed.

Precision and recall are common metrics for the evaluation of prediction and classification models [37]. Precision is the percentage of correct detections. It measures the ability of a model to select only desired objects. On the other hand, recall is the percentage of found objects. It shows how well a model can find all desired objects in a dataset. These metrics are usually used together. Apart, they hold limited value. Individually, each metric can achieve a high score. High precision can be achieved by only accepting weed detections of extremely high certainty and removing the rest. High recall can be attained by detecting every plant in the image. Therefore, precision and recall are evaluated together and usually have an inverse relationship (as the number of found objects increases, so does the number of false positives, thus decreasing the precision). This trade-off between precision and recall is depicted using a precision–recall curve (PRC). The curve is a plot of precision as a function of recall. As the object detection thresholds are relaxed and more objects are found in the image (increasing recall until all are found), the progression of precision is mapped to it. A good detector will maintain high precision levels for longer.

The mAP50 stands for mean average precision at a 50% intersection over union (IoU). It is often used as a main metric in object detection studies (e.g., the Pascal Visual Object Classes Challenge [36]). The mean average precision is calculated using the PRC. The recall value range (from 0 to 1) is divided into 11 points, and precision is calculated for each of them. The mean of these values creates the mAP metric. The IoU is a bounding box overlap metric. The ground truth object bounding boxes and predicted bounding boxes are compared, and the overlap percentage is calculated. For the calculation of mAP50, only detected objects with at least 50% overlap are considered positive.

The inference speed is the amount of time a model needs to process one image. It is tied to the complexity of the models. The main reason for slower inference is the number of parameters (number of weights and biases) that are included in the model.

3.6. Hardware

The object detection training was performed using Google Colab service with 12 GB RAM, NVIDIA Tesla T4 GPU (Nvidia Corporation, Santa Clara, CA, USA) and CUDA version 12.2. Inference speed testing was carried out using Raspberry Pi 5 (RPi5; Raspberry Pi Foundation, Cambridge, UK) with 8 GB RAM and Jetson Nano (Nvidia Corporation). The RPi5 was chosen as a lightweight CPU-based tool for processing image data which can be used on small mowers. The RPi platform has previously been used in weed detection studies [38,39]. While lacking a dedicated graphics card, its power consumption is considerably lower than some of the computer vision kits (e.g., Zed Box—5A19V vs. 5A5V). Jetson Nano is another example of a single-board computer. Unlike RPi5, it has a dedicated GPU, which improves its DL processing times. Jetson Nano consumes 10 W by default but can be switched to a 5 W power-saving mode.

The following software packages were installed:

- Python 3.11;
- YOLOv8.1.2;
- PyTorch 2.2.2;
- CUDA 12.1 (Jetson only);
- Additional dependencies.

4. Results

4.1. Model Accuracy

Table 2 provides an overview of the precision, recall and mAP50 metrics across different models. For the v8n model, the mAP50 scores of RGB (0.557) and RGD (0.56) were closely tied. However, RGD had higher precision and recall scores, suggesting that the boundary box estimates were poorer. When examining the PRC for YOLOv8n (Figure 2), RGB and RGD both started similarly. In the middle ranges, the RGB suffered a drop in precision, but it performed better at higher recall values. The RGD had the opposite tendency. Therefore, RGD might perform better in cases where the detection thresholds are higher. The DGB variant had the highest precision score of 0.632 but suffered from the worst recall of all with 0.512, implying that while it can correctly reject erroneous object proposals, it struggled with finding all weed instances in images. Both DGB and RDB had similarly low mAP50 scores: 0.533 and 0.532.

The YOLOv8s results (Figure 3) differed from those of YOLOv8n. In this case, the RGB was the worst-performing band combination in all metrics. The PRC showed that it was outperformed by all other versions, with a recall of 0.3 to 0.7. The best performance for this model was that of the RDB band version. Its mAP50 score was increased by 8% compared to RGB (0.621 vs. 0.574), precision by 13% (0.635 vs. 0.562) and recall by 9% (0.572 vs. 0.523). Its PRC also showed a solid performance in all ranges. In this model, RGD came second, with an mAP50 of 0.596. Its recall of 0.619 was higher than that of RDB but suffered from the second lowest precision score of 0.586.

In the case of YOLOv8m, RDB had the highest mAP50 score of 0.634 and a precision of 0.692. Its PRC (Figure 4) showed a solid performance until the recall value of 0.7, where it fell in line with RGB. The RGB was the second best performing this time, with an mAP of 0.613, which is only 3% behind RDB. The precision score (0.64) was 7.5% lower than that of RDB. Interestingly, RGD showed a much higher recall value compared to the other band variants. At 0.632, it was 11% higher than the next value of 0.566 for RGB and 23% higher than the lowest value of 0.512 for DGB.

Table 2. Precision, recall and mAP50 metrics for all band combinations and selected models.

Model	Bands	Precision	Recall	mAP50
v8n	RGB	0.565	0.53	0.557
	DGB	0.632	0.512	0.533
	RDB	0.566	0.557	0.532
	RGD	0.624	0.562	0.56
v8s	RGB	0.562	0.523	0.574
	DGB	0.615	0.564	0.592
	RDB	0.635	0.572	0.621
	RGD	0.586	0.619	0.596
v8m	RGB	0.64	0.566	0.613
	DGB	0.639	0.512	0.562
	RDB	0.692	0.562	0.634
	RGD	0.561	0.632	0.573

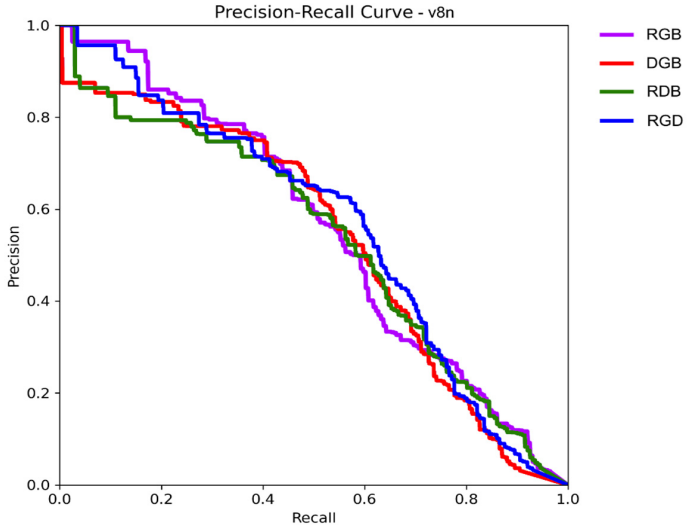


Figure 2. Precision–Recall Curve for v8n model.

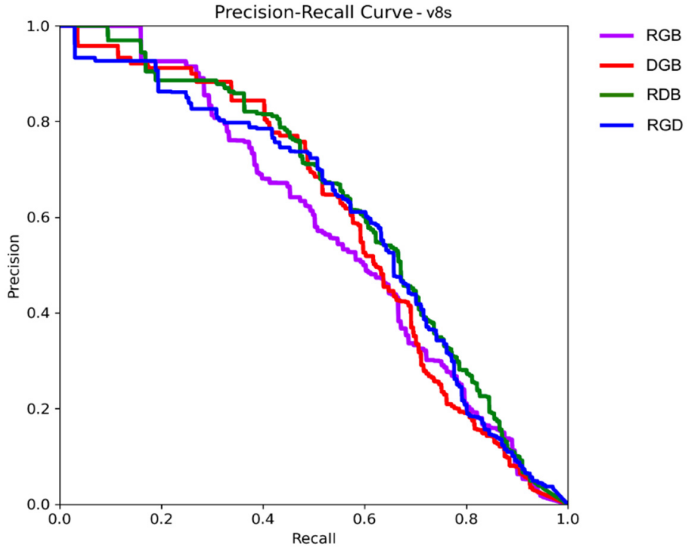


Figure 3. Precision–Recall Curve for v8s model.

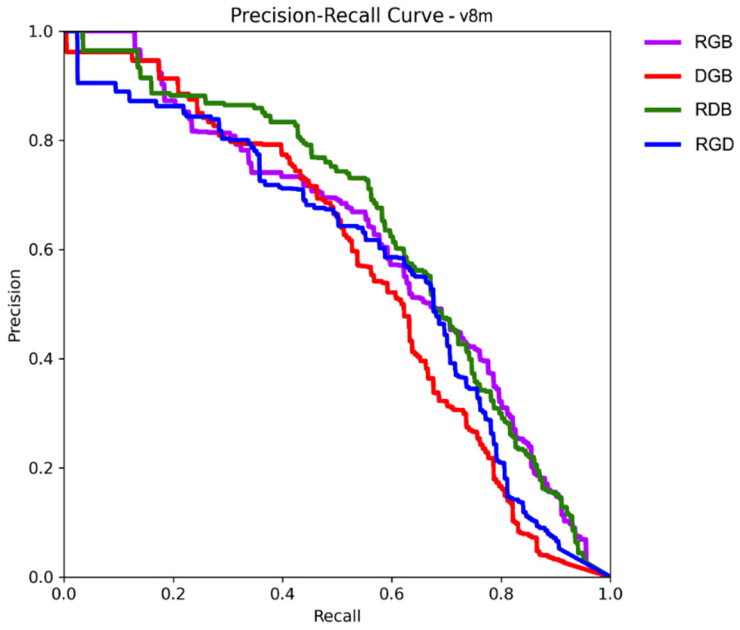


Figure 4. Precision-Recall Curve for v8m model.

4.2. Inference Speed

The evaluation of inference times on RPi5 and Jetson Nano is provided in Table 3. The measured speed is in milliseconds. The number is the average of the inference times of all images in the testing dataset.

Table 3. Average inference speed in milliseconds for RPi5 and Jetson Nano single-board computers.

Model	Speed (ms)	
	RPi5	Jetson Nano
v8n	431	176
v8s	808	342
v8m	1835	758

The RPi5 performs well with the v8n model, only needing 431 ms to perform inference. This speed would allow for a continual survey of the fields without pause. While v8n has the worst performance in terms of accuracy, it might be considered in cases where the field robot is constructed with higher movement speed in mind. Since the inference time of the next model (v8s) is 808 ms—an increase of 87%—the trade-off in accuracy can be acceptable. However, depending on the field robot design, camera placement, field of view and operating speed, v8s can also be considered viable. As long as there is enough overlap between consecutive images to avoid detection errors from a lack of data, the increase in accuracy of v8s over v8n will provide better weeding results. The v8m model, on the other hand, is not practical for field operations with a 1835 ms processing time. If higher accuracy is critical for field operations, different hardware should be considered.

As expected, the dedicated GPU of Jetson Nano allows for faster object detection processing times than RPi5. The v8n model inference is only 176 ms. That is a 59% decrease in the required time. Considering the movement speed of small field robots, the performance can be counted as real-time processing. The v8s processing time is also fairly low. At 342 ms, it is 26% faster than the performance of RPi5 with the v8n model and can be considered fully viable. The v8m model is more cumbersome, with 758 ms. However, in terms of performance, it is still 50 ms faster than the RPi5 using v8s.

The choice between RPi5 and Jetson Nano also comes down to project requirements, practicality, user experience and choice of peripherals. For robots with higher operating movement speeds, Jetson Nano should be considered. However, RPi has much wider support for sensors and other peripherals, as well as a larger user base and support.

5. Discussion

This study showed there is potential in depth band substitution. Based on the results, it appears the RDB combination could be beneficial for weed detection in pastures. It proved to be a superior option in two out of the three tested models. While its mAP50 score in the v8n model was lower, it had better precision and recall scores than RGB. This points to an issue with correctly localizing plants, which could potentially be attributed to insufficient training data. The dissimilarities in plant stages of broad-leaved dock could also play a role. Some studies expand the roster of weed classes with different plant stages [40]. Such a distinction might improve future analyses.

The performance of detection across the tested models is only significant between v8n and v8s. The difference between the best-performing band variants (RGD and RDB) is 6.1%. However, when comparing the performance of RDB in v8s and v8m, the change is only 1.3%. On RPi5, the switch from v8s to v8m would require over a second of additional time to process, while Jetson Nano would need an extra 416 ms. Considering the increase in computational cost, it seems that v8m is not a practical option.

Future studies should focus on expanding this comparative analysis to other object detectors. Different architectures could have divergent responses to the depth band substitution. A more receptive model may exist. Furthermore, additional image data would facilitate a more comprehensive analysis. Another avenue of research is determining how image augmentation changes for images with a depth band. Color transformation especially might have a different effect than with classic RGB imagery. Finally, the question is whether a different product of depth vision might prove more beneficial. In RGBD object detection, disparity and angle data are also used. One of these could provide better results than depth. Potentially, a combination of two depth products with a single color band could also be explored.

Author Contributions: Conceptualization, J.V. and A.K.; methodology, J.V.; software, J.V. validation, J.V. and J.P.; formal analysis, J.V.; investigation, J.V., A.K. and J.P.; resources, J.P. data curation, J.P.; writing—original draft preparation, J.V.; writing—review and editing, J.V., A.K. and J.P. visualization, J.V.; project administration, A.K.; funding acquisition, A.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Swiss National Science Foundation, Practice-to-Science, grant number PT00P2_199063.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data associated with the work described in this publication are available at the Dryad data repository [23].

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

References

1. Filly, S. Nutritional Value of Pasture Forages Changes Through the Growing Season. Oregon State Extension Service. 2024. Available online: https://extension.oregonstate.edu/crop-production/pastures-forages/nutritional-value-pasture-forages-changes-through-growing-season (accessed on 13 May 2024).

2. Van Evert, F.K.; Cockburn, M.; Beniers, J.E.; Latsch, R. Weekly defoliation controls, but does not kill broad-leaved dock (Rumex obtusifolius). *Weed Res.* **2020**, *60*, 161–170. [CrossRef]
3. Shanmugam, S.; Assunção, E.; Mesquita, R.; Veiros, A.; Gaspar, P.D. Automated Weed Detection Systems: A Review. *KnE Eng.* **2020**, *5*, 271–284. [CrossRef]
4. Wang, A.; Zhang, W.; Wei, X. A review on weed detection using ground-based machine vision and image processing techniques. *Comput. Electron. Agric.* **2019**, *158*, 226–240. [CrossRef]
5. Hasan, A.S.M.M.; Sohel, 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]
6. Ariza-Sentís, M.; Vélez, S.; Martínez-Peña, R.; Baja, H.; Valente, J. Object Detection and Tracking in Precision Farming: A Systematic Review. *Comput. Electron. Agric.* **2024**, *219*, 108757. [CrossRef]
7. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep Learning in Agriculture: A Survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]
8. Masoumian, A.; Rashwan, H.A.; Cristiano, J.; Asif, M.S.; Puig, D. Monocular Depth Estimation Using Deep Learning: A Review. *Sensors* **2022**, *22*, 5353. [CrossRef]
9. Hou, S.; Wang, Z.; Wu, F. Object Detection via Deeply Exploiting Depth Information. *Neurocomputing* **2018**, *286*, 58–66. [CrossRef]
10. Takahashi, M.; Ji, Y.; Umeda, K.; Moro, A. Expandable YOLO: 3D Object Detection from RGB-D Images. In Proceedings of the 2020 21st International Conference on Research and Education in Mechatronics (REM), Cracow, Poland, 9–11 December 2020; IEEE: Cracow, Poland, 2020; pp. 1–5.
11. Meyer, G.E.; Neto, J.C. Verification of Color Vegetation Indices for Automated Crop Imaging Applications. *Comput. Electron. Agric.* **2008**, *63*, 282–293. [CrossRef]
12. Sabzi, S.; Abbaspour-Gilandeh, Y.; García-Mateos, G. A Fast and Accurate Expert System for Weed Identification in Potato Crops Using Metaheuristic Algorithms. *Comput. Ind.* **2018**, *98*, 80–89. [CrossRef]
13. Slaughter, D.C. The Biological Engineer: Sensing the Difference Between Crops and Weeds. In *Automation: The Future of Weed Control in Cropping Systems*; Young, S.L., Pierce, F.J., Eds.; Springer: Dordrecht, The Netherlands, 2014; pp. 71–95. ISBN 978-94-007-7511-4.
14. Ji, R.; Qi, L. Crop-Row Detection Algorithm Based on Random Hough Transformation. *Math. Comput. Model.* **2011**, *54*, 1016–1020. [CrossRef]
15. Hasan, A.S.M.M.; Diepeveen, D.; Laga, H.; Jones, M.G.K.; Sohel, F. Object-Level Benchmark for Deep Learning-Based Detection and Classification of Weed Species. *Crop Prot.* **2024**, *177*, 106561. [CrossRef]
16. 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]
17. 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]
18. 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]
19. Cao, Y.; Shen, C.; Shen, H.T. Exploiting Depth from Single Monocular Images for Object Detection and Semantic Segmentation. *IEEE Trans Image Process.* **2017**, *26*, 836–846. [CrossRef]
20. Chu, F.; Pang, Y.; Cao, J.; Nie, J.; Li, X. Improving 2D Object Detection with Binocular Images for Outdoor Surveillance. *Neurocomputing* **2022**, *505*, 1–9. [CrossRef]
21. CABI Digital Library, Rumex Obtusifolius (Broad-Leaved Dock). 2022. Available online: https://www.cabidigitallibrary.org/doi/full/10.1079/cabicompendium.48064 (accessed on 18 December 2024).
22. Stereolabs ZED 2i Datasheet. Available online: https://cdn.sanity.io/files/s18ewfw4/production/f3860c2dfd475deb15f6d5ed28d2b106d6639d94.pdf/ZED%202i%20Datasheet%20Feb%202022.pdf (accessed on 2 January 2024).
23. Vandrol, J.; Perren, J.; Koller, A. RGBD Data of Broad-Leaf Dock. *Data Dryad* **2024**. [CrossRef]
24. Tkachenko, M.; Malyuk, M.; Holmanyuk, A.; Liubimov, N. Label Studio. Available online: https://github.com/heartexlabs/label-studio (accessed on 18 December 2024).
25. Wang, W.; Chen, J.; Huang, Z.; Yuan, H.; Li, P.; Jiang, X.; Wang, X.; Zhong, C.; Lin, Q. Improved YOLOv7-Based Algorithm for Detecting Foreign Objects on the Roof of a Subway Vehicle. *Sensors* **2023**, *23*, 9440. [CrossRef]
26. Bunker, J.; Hadjidemetriou, G.M.; Marie d'Avigneau, A.; Girolami, M. On the Performance of Pothole Detection Algorithms Enhanced via Data Augmentation. *Transp. Res. Procedia* **2024**, *78*, 230–237. [CrossRef]
27. Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and Flexible Image Augmentations. *Information* **2020**, *11*, 125. [CrossRef]
28. Jin, X.; Sun, Y.; Che, J.; Bagavathiannan, M.; Yu, J.; Chen, Y. A Novel Deep Learning-based Method for Detection of Weeds in Vegetables. *Pest Manag. Sci.* **2022**, *78*, 1861–1869. [CrossRef]
29. Ajayi, O.G.; Ashi, J.; Guda, B. Performance Evaluation of YOLO v5 Model for Automatic Crop and Weed Classification on UAV Images. *Smart Agric. Technol.* **2023**, *5*, 100231. [CrossRef]

30. 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]
31. Rai, N.; Zhang, Y.; Villamil, M.; Howatt, K.; Ostlie, M.; Sun, X. Agricultural Weed Identification in Images and Videos by Integrating Optimized Deep Learning Architecture on an Edge Computing Technology. *Comput. Electron. Agric.* **2024**, *216*, 108442. [CrossRef]
32. Wang, Q.; Cheng, M.; Huang, S.; Cai, Z.; Zhang, J.; Yuan, H. A Deep Learning Approach Incorporating YOLO v5 and Attention Mechanisms for Field Real-Time Detection of the Invasive Weed Solanum Rostratum Dunal Seedlings. *Comput. Electron. Agric.* **2022**, *199*, 107194. [CrossRef]
33. 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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
34. Jocher, G.; Chaurasia, A.; Qiu, J. Ultralytics YOLO Vision. Available online: https://github.com/ultralytics/ultralytics (accessed on 18 December 2024).
35. Dyrmann, M.; Karstoft, H.; Midtiby, H.S. Plant Species Classification Using Deep Convolutional Neural Network. *Biosyst. Eng.* **2016**, *151*, 72–80. [CrossRef]
36. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
37. Davis, J.; Goadrich, M. The Relationship between Precision-Recall and ROC Curves. In Proceedings of the 23rd International Conference on Machine Learning—ICML '06, Pittsburgh, PA, USA, 25–29 July 2006; ACM Press: Pittsburgh, PA, USA, 2006; pp. 233–240.
38. Chechlinski, L.; Siemiątkowska, B.; Majewski, M. A System for Weeds and Crops Identification—Reaching over 10 FPS on Raspberry Pi with the Usage of MobileNets, DenseNet and Custom Modifications. *Sensors* **2019**, *19*, 3787. [CrossRef]
39. Mishra, S.; Sachan, R.; Rajpal, D. Deep Convolutional Neural Network Based Detection System for Real-Time Corn Plant Disease Recognition. *Procedia Comput. Sci.* **2020**, *167*, 2003–2010. [CrossRef]
40. Wu, Z.; Chen, Y.; Zhao, B.; Kang, X.; Ding, Y. Review of Weed Detection Methods Based on Computer Vision. *Sensors* **2021**, *21*, 3647. [CrossRef] [PubMed]

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.