Augmentation of sunflower-weed segmentation classification with Unity generated imagery including near infrared sensor data

Carlos Carbone[1[0000−1111−2222−3333]], Ciro Potena[1[1111−2222−3333−4444]], and Daniele Nardi[1[2222−−3333−4444−5555]]

Sapienza University of Rome, Via Ariosto 25, Italy

Abstract. This paper presents a solution to create synthetic datasets for deep learning training of convolutional neural networks (CNNs) for plant-weed classification. We use the Unity game engine to create simulated procedural fields of sunflowers and weeds images. The visual imagery is generated by the photo realistic real time rendering engine in Unity. Moreover, we include the regular red green and blue (RGB) channels plus the near infrared (NIR) channel data by including the aligned textures from both the RGB and the NIR channel separately since Unity doesn't simulate NIR illumination.
Our main contribution is the simulation of the sunflower plant including both the RGB and the NIR data based of an real image dataset with low quality and quantity to generate improved datasets that can reliably train CNNs for plant-weed segmentation classification. The results obtained achieve high intersection over union (IoU) performance when we build a dataset including a small subset of the synthetic images with the high amount of plant and weed pixel data plus the available real images for training.
Our best results show an IoU performance of 76.4% training the CNN only with sunflower synthetic images. This is close to the results from our previous research where the available real dataset of sugar beets had ideal conditions of quality and quantity. Therefore, we conclude that using synthetic imagery including both RGB and NIR data can greatly improve plant-weed segmentation classification IoU performance when the real images available have limited quality and quantity.

Keywords: Convolutional Deep Learning · Precision Agriculture · Unity.

1 Introduction

This paper serves as an extension for the paper titled "Simulation of near infrared sensor in Unity for plant-weed segmentation classification" [9] presented in the SIMULTECH 2020 conference. It presented the novel simulation of the near infrared sensor in the Unity game engine to generate training datasets for segmentation classification of sugar beets and weeds. In this paper we present results for sunflower and weeds using a dataset with lower quality and quantity to prove the performance of the presented method with alternative non ideal conditions while still achieving a high and reliable intersection over union (IoU) performance.

1.1 Segmentation classification in precision agriculture

The International Society of Precision Agriculture establishes the following definition: "Precision Agriculture is a management strategy that gathers, processes and analyzes temporal, spatial and individual data and combines it with other information to support management decisions according to estimated variability for improved resource use efficiency, productivity, quality, profitability and sustainability of agricultural production." [19].

The main objective of precision agriculture practices is to provide more efficient farming methods for the food demand coming from the increasing earth population [1]. Moreover, difficulties for farmers increase with the common trend of workers leaving the country side to pursue more promising opportunities in modern urban cities [28, 2]. Thus, robotics research for precision agriculture has been on the rise in the recent years [42, 22, 36, 14, 11, 9].

The robotics solutions provided for precision agriculture require sensor data to be accurately processed for the farmers to take precise decisions or for other robots to do any physical activity required in the crop fields. One challenge of interest lies in accurate weed recognition and localization, this provide the required data to save resources in chemicals and workforce for weed removal to increase crop yield quality [12]. In particular, convolutional neural networks (CNNs) is providing promising results as a solution for weed detection through imagery sensor data [26, 38]. However, these CNNs require to be trained using datasets of images with high quality and quantity that are properly labeled to achieve reliable image segmentation classifications [8, 9, 13, 29]. Thus, in our previous paper we provided a solution using the photo realistic real time rendering from a video game engine to quickly generate imagery that serves to build reliable datasets for CNN deep learning training. We used the Unity game engine to generate a sugar beet dataset with the addition of the near infrared (NIR) sensor data which wasn't done before. This provided an improvement for the intersection over union (IoU) evaluation of the segmentation classification performance, especially for the case where training was based exclusively on synthetic generated images [9].

Our main contribution in this extended paper is the addition of the sunflower plant to the Unity environment, including the red-green-blue (RGB) and NIR channel, based on a dataset that has low quality and quantity to prove the capability of the solution in limited conditions while still achieving high IoU performance. Thus, the rest of the paper is presented as follows: Section 2 presents the state of the art mainly for video game engine simulations and the CNN used for this paper, Section 3 presents the base dataset used for this research, Section 4 presents how the textures were extracted to crate the Unity procedural level, Section 5 presents the dataset requirements for the selected CNN to be trained and how these datasets were built, Section 6 shows the results and their discussion, and finally Section 7 presents the conclusions obtained from the result behaviors.

2 Related work

This section presents the state of the art with related works to our research. First we present researches for synthetic data generation for machine learning and CNN research purposes using video games and video game engines approaches as they have taken a considerable prominence in the recent years. Our solution in this paper follows and exploit this trend as well. Then we present CNNs researches within precision agriculture context including the selected CNN for our tests.

Simulations in video games and video game engines have been considered a data source for machine learning research purposes [39]. Video games are capable of producing large amounts of pixel and command data, which is highly valuable for this type of research [44, 20]. However, video game engines are also taking prominence in the research field as video games are constrained to the game hard coded rules, thus creating new environments with custom graphics and behaviors becomes a key factor to create relevant and innovative researches [23]. Their photo realistic imagery provided by their real time rendering engines makes them an attractive solution to easily and safely generate data resembling real world problems [9, 18].

The most used video game engines are Unity3D [43], hereinafter referred to as "Unity", and Unreal Engine 4 [4], hereinafter referred to as "Unreal", because of their promising imagery available for free to the users. Both engines are in constant development surpassing each other in different metrics, but one trend that seems to be constant is that Unreal seeks maximum photo realism while Unity focus on providing a high level of compatibility [7, 35].

The video game engines have taken prominence in the recent years for self driving car simulations [16, 24, 21, 5]. In [17] a visually and physically realistic simulation scenario in Unity is proposed to generate the required data to train a CNN for a self driving shuttle. In [6] a pedestrian-vehicle environment is built to generate synthetic images to train a neural network through UnrealCV. In [45] Unity is used for car simulations to study sensor reaction behaviors. In [31] over 480 000 labeled images were generated from the game Grand Theft Auto V from highway driving scenes which provided training data to achieve good estimates from the driver's perspective to lane marking, distance to cars and angle of driving. These results were compared to a Unity developed environment that was able to test certain scenarios that were not possible by the constraint game environment of Grand Theft Auto V.

In [37] Unreal was used to generate multiple views of the Soyuz spacecraft to train the GoogLeNet CNN to determine its position and orientation. The results provided a position accuracy of 92.53 percent with an error of 1.2 m and an maximum orientation error of 7.6 degrees.

In [32] Unreal was used to create an image dataset to train a neural network to recognize deformed objects. The results achieved a correct recognition between 50 and 60 percent.

The synthetic data generation for training of CNNs in the agriculture context remains considerably unexplored as to the best of our knowledge our presented paper in [9] was the first to include the NIR data for this type of simulations. It is worth noting that this data is key in precision agriculture as it is one of the measurements that directly represents plant health through the normalized difference vegetation index (NDVI). Additionally, it increases performance of CNN for segmentation classification [34].

In [13] sugar beets were simulated but only including the RGB channels using Unreal since the main focus was to achieve a high level of photo realism for crop-weed classification, achieving 61.1% for the mean IoU evaluation using only synthetic data [13, 9].

In our previous research in [9], we presented the near infrared simulation using Unity since we required our tools to be widely available to other users, including Linux and robotic operating system (ROS) users for which Unreal presented lesser ease of use. The plant selected for simulation was the sugar beet and considerable effort went into gathering good quality of real images to extract the textures for plant simulation. We selected sugar beets because of the large dataset available from the 2016 Sugar Beets Dataset Recorded at Cam-pus Klein Altendorf in Bonn [12] which we used for comparison purposes. Additionally, it served to make a comparison with the performance form the Unreal synthetic dataset which presented superior traits for the RGB segmentation classification IoU, however a different CNN was used.

2.1 CNNs in precision agriculture

As mentioned, the main purpose of training CNNs for precision agriculture is to identify the location of plants and weeds in the field crop required to enhance decision making for farmers and robots [29, 10]. One of the main solutions to collect crop image nowadays is the use of unmanned aerial vehicles (UAV), which can carry sensors while flying over the crops, thus quickly gathering large amount of high quality images [25, 10]. Thus, in [40] it was developed a system for reliable plant-weed classification based on UAV hardware limitations. The Jetson TX2 was integrated in the UAV system to be able to handle SegNet. Then, in [41] further efforts were made to include multispectral sensor data in the UAV CNN training. In [30] a different approach is made to instead decrease the computational requirements of a CNN for farming robots. They exploit the repetitive structure of crop fields achieving labeling times of around 1 minute with an accuracy of more than 95% for sugar beets.

In [29] presents a fully convolutional network with an encoder-decoder structure. This incorporates spatial information by considering images sequences, thus the observable crop arrangement is exploited following the research in [30]. The new system provides an accuracy improvement for crop-weed classification in unseen fields without the need of retraining the model.

The main CNN we chose for this research was developed in [33] which was tested for multiple types images including plant-weed segmentation classification with sugar beets. It provided, a 80.1% for the mean IoU and 98.5% for the mean accuracy using the dataset from [12]. More details about its setup are mentioned in Section 5.

It is important to note that other approaches have been studied to achieve dataset augmentation without simulation procedures. In [15] the dataset images were rotated, scaled and mirrored to generate more images variation to extend the training quality. Then the research was focused on developing an architecture of two networks, the first classifies connected patches of plant from the soil, then the second makes the label classification. Furthermore, in [27] an approach to overcome dense weeds and overlap with plants is presented using ResNet-10 including the Adaptative Affinity Fields method.

3 Sunflower dataset

The main purpose in this research is to study the support that this method can provide in situations with limited resources. We used a sunflower dataset with 146 images of 1296x964, including the RGB, NIR and labeled versions of each image. From these, 105 images were taken for classification evaluation, 34 for a training dataset, and 7 for texture extraction to simulate the sunflower plant and weeds in Unity. Additionally, no camera calibration matrix was available, therefore the alignment between RGB, NIR and the labeled version was not perfect. The brightness exposure in the images was also high, making some of the leaf features considerably uniform and hard to distinguish, especially in the NIR channel which can crate confusion with some of the rocks in the ground as shown in Figure 1.

This creates a difficult scenario to perform good CNN deep learning training. Therefore, the addition of synthetic images generated with Unity can provide extra support to achieve higher levels of classification accuracy for the trained network.

4 Simulated crop

In this research we simulated a sunflower field crop in Unity, including weeds. Similarly to our previous research, we extracted the textures from the dataset images taken with a JAI camera sensor that includes the RGB and NIR channels. However, in this paper the main purpose is to study the support that this method can provide in situations with limited resources. Thus, only 7 images were used images to extract the textures for both the sunflower plants and the weeds. With these few images, multiple random variations were created for both the sunflower plants and weeds.

4.1 Unity level

First, the weeds are extracted using the same method from the previous research. The labeled version of the images were used to isolate the weeds creating 7 weed textures which were included in simple quads during the crop procedural simulation, Figure 2 shows an example.

The sunflower plant textures where extracted using the gimp image editor software and the labeled version of the images [3]. However, a few adjustments by hand were needed since the labeled version of the images included some pixels from the ground behind the plants. The sugar beet leaves are homogeneous enough for the textures to be included in randomly bent virtual leaves in our previous paper. For this case, the sunflower structure is more sophisticated since the leaves grow in a repetitive pattern where two leaves grow out of a tiny "mouth" that comes from the center of the previous two leaves, each pair is rotated in approximately 90 degrees as shown in Figure 1. Therefore the procedural code required the leaf textures to be specific for the plant structure section dividing the leaves: in the lower "root" leaves, upper "head" leaves, and "Center" leaves. Figure 3 shows the textures obtained by cropping the leaves from one of the real images available, including its RGB and NIR version, the masked version is obtained directly in Unity by using shaders with plain color. The leaves were not centered inside the texture images to match the 3D leaf UV mapping which is the system 3D graphics engines use to map texture pixels over 3D meshes. The center leaves are centered since they were added in quad polygons instead for simplicity purposes.

The obtained textures were added to the 3D meshes used for the sugar beet in our previous research since they were simple rectangular meshes that were further bent. Figure 4 shows how the obtained textures are placed over the 3D mesh through the shading system of Unity where a the transparency from the texture is taken into account for the mesh rendering.

The sunflowers from the dataset where at a young state when the weeds grow, thus the simulated structure needs to be equivalent. The plant stem is not simulated for simplicity purposes since it is not visible in the images from the dataset and the leaves were divided into head leaves, root leaves and center leaves (the small "mouth" on top of the plant) as mentioned. The variations included: 6 root leaves variations, 6 head leaves variations and 3 center leaves variations. Figure 5 shows a comparison between a synthetic generated image and a real image, comparing both RGB and NIR versions. Then, Figure 6 shows one procedural field generated for image extraction, including the RGB, NIR and Masked versions which changes the textures by plain colors to server as labels for the deep learning training.

The procedural field provides several random variations for as many images as required, thus 1000 images including each version (RGB, NIR and labeled) were generated, similarly to our previous research where 1034 sugar beet images where generated. Next section shows how these were distributed for the CNN deep learning training.

5 CNN deep learning training

The CNN used for this research is "Bonnet: An Open-Source Training and Deployment Framework for Semantic Segmentation in Robotics" [33], hereinafter referred to as "Bonnet". This network was designed partially for plant-weed segmentation classification providing very promising results. Bonnet requires the datasets to be distributed into three subsets named train, validation, and test.

5.1 Dataset distribution

The synthetic and real datasets distribution is summarized in Table 1. The rows show the dataset image dimensions and the amount of images per subset for Bonnet. The image dimensions were based on: The JAI camera resolution for the real images, the common image resolution for a power of two texture in a game engine for the synthetic images and a smaller image dataset, and a smaller power of two for the augmented case to combine both type of images into a common dataset.

The columns show the datasets generated. The *Real* and *Unity* dataset were built using all the real and synthetic images available for training respectively and the *Augmented* dataset is the combination of both datasets. Then the *Reduced* dataset is made of a total of 34 Unity synthetic images to match the amount of images from the real dataset for a direct performance comparison. Finally the *Reduced augmented* dataset is the combination of the *Real* dataset and the *Reduced* dataset.

Table 1: Sunflower datasets features.

Feature for RGB and RGBN input	Real	Unity	Augmented	Reduced	Reduced augmented
Dimensions (WxH pixels)	1296x964	1024x1024	512x512	1024x1024	512x512
Train (number of images)	700	25	700 Unity + 25 real	25	25 Unity + 25 real
Validation (number of images)	150	5	150 Unity + 5 real	5	5 Unity + 5 real
Test (number of images)	150	4	150 Unity + 4 real	4	4 Unity + 4 real

Table 2 shows the dataset distribution used in our previous research for the sugar beet CNN training, where our main focus was to evaluate the addition of the NIR in ideal conditions, that is, having enough real images for training and classification. In that case the real and the Unity datasets each had 1034 images and the "Augmented" case which is the combination of both had the 1034 images plus 300 real images. This left another 820 additional real images for classification evaluation [9].

Table 2: Sugar beet datasets features.

Feature for RGB and RGBN input	Real	Unity	Unity + Real
Dimensions (WxH pixels)	1296x966	1024x1024	512x512
Train (number of images)	734	734	734 Unity +100 real
Validation (number of images)	150	150	150 Unity +100 real
Test (number of images)	150	150	150 Unity +100 real

5.2 Evaluation method

The main key performance indicators (KPI) to evaluate the presented method are the intersection over union (IoU) and the accuracy. These are measurements for the segmentation classification performance which are applied to each class label: plants, weeds and ground. Then, a mean is calculated to obtain an overall evaluation.

The accuracy is calculated using Equation 1, where T_{pi} is a true positive for a pixel i, meaning it is classified correctly for a given label (a plant pixel is classified as a plant pixel). Then, F_{pi}, F_{ni} and T_{ni} are the false positive, false negative and true negative respectively. The true negative is the outcome when a pixel is classified correctly as not a given class, for example, a ground or weed pixel is not classified as a plant pixel.

The sum of the four types of outcomes gives all the pixels of an image, thus the accuracy is obtained by summing all the true positives and true negatives and dividing the result by the sum of all the pixels of the image.

$$Accuracy_{label} = \sum_{i=1}^{N} \frac{T_{pi} + T_{ni}}{T_{pi} + T_{ni} + F_{pi} + F_{ni}} \tag{1}$$

The IoU is calculated using Equation 2, where the main difference with the accuracy is that the true negatives are not taken into account. Using mainly the true positives as the main variable to determine if the classification is good. This means that a plant IoU results will be high only if the plant pixels are correctly classified as plants, the same being the case for the rest of the class labels.

$$IoU_{label} = \sum_{i=1}^{N} \frac{T_{pi}}{T_{pi} + F_{pi} + F_{ni}} \tag{2}$$

Usually the IoU takes more prominence within the evaluation of a segmentation classification method since it is directly proportional only to the true positives while the accuracy results can also increase with the true negatives. This is a key factor since it is possible to obtain high accuracy because true negatives are not likely to be low, thus, increasing the accuracy result. For example, is not very likely that the field ground will be classified as a plant, thus even if the true positives to classify the plant are not good, the final accuracy will still be very high since there are a lot more background pixels than plant pixels. This means that to obtain a high IoU the classification must correctly classify the pixels for a given class independently of the pixels belonging to other classes.

The results obtained the IoU and accuracy for each class label are multiplied by 100 to present them in percentage. Then, the mean between the three classes is calculated for an overall result.

6 Results

This section presents the results for the segmentation classification after the training Bonnet using the datasets presented in Table 1.

Table 3 shows the mean result over the classification of 105 sunflower real images for each class and for the mean calculated over the classes in each image. Additionally we include Table 4 from our previous paper including mean result for the classification over 820 images using the datasets presented in Table 2, the IoU graph for these results is presented in Figure 9 [9].

In the presented tables, the results for the classification over each class label is presented, including results using the RGB data and using the RGB and NIR data which we address as RGBN. The tables show in the first rows the IoU over each class and the mean, then it shows the accuracy over each class and the mean in the second half of the rows.

The columns represent the datasets used during the CNN deep learning training, the *Real* dataset with all the available real images for training, the *Unity* dataset with all the available synthetic images for training. Then, the *Augmented* case which we previously called "Unity + Real" is the combination of both, which was all the available synthetic images + 300 real images, however for the sunflower case we used all of the available ones since the amount available (34) was low. Then the Reduced and Reduced augmented dataset uses only 34 synthetic images and 34 synthetic plus 34 real images respectively.

6.1 Discussion

The results presented in Table 3 and Figure 7 show that the best mean performance come from the classification with the CNN trained with the *Reduced Augmented* dataset. It is worth mentioning that the images selected for the *Reduced* dataset were carefully hand picked to include good quality images that contain good amounts of pixel data for plants and weeds. The ground was not greatly considered for this assessment since the texture has a very low level of variation because of the focus on plant and weed classification.

In the IoU classification results, the addition of the NIR data to the training datasets brought an improvement for the mean result for the performance from each dataset training. However, the individual class classification for the sunflower plant was affected negatively by the addition of the NIR channel for some of the training datasets, this is can be mainly due the high brightness mentioned for Figure 1 where some of the features get lost by turning fully white similar to some rocks from the ground.

Table 3: Classification mean results (in percentage) for classification over 105 sunflower images used for evaluation performance.

Method	Real	Unity	Augmented	Reduced	Reduced Augmented
Plant IoU RGB	70.34	85.02	78.05	78.96	78.97
Plant IoU RGBN	65.74	83.45	81.77	76.84	80.3
Weed IoU RGB	38.57	44.77	38.36	41.49	47.91
Weed IoU RGBN	43.85	47.43	54.88	53.98	57.92
Ground IoU RGB	98.04	98.13	97.17	98.22	98.01
Ground IoU RGBN	98.53	97.97	98.61	98.37	98.72
mean IoU RGB	68.98	75.97	71.19	72.89	74.96
mean IoU RGBN	69.98	76.29	78.42	76.4	78.98
Plant Accuracy RGB	99.15	99.76	99.69	99.68	99.72
Plant Accuracy RGBN	99.42	99.76	97.83	99.66	99.76
Weed Accuracy RGB	97.92	98.50	97.83	98.42	98.48
Weed Accuracy RGBN	99.59	98.70	98.87	99.99	99.99
Ground Accuracy RGB	99.98	99.99	99.99	99.99	99.99
Ground Accuracy RGBN	99.98	99.99	99.99	99.99	99.99
mean Accuracy RGB	99.02	99.42	99.17	99.37	99.39
mean Accuracy RGBN	98.66	99.49	99.54	99.45	99.57

The most surprising result was the high IoU for the CNN trained with the *Reduced* and *Reduced Augmented* datasets. This can be mainly due the possible noise that comes with the randomness with the Unity simulation, thus the best performance for this case comes from manually selecting the best images from from the *Unity* dataset to create the *Reduced* dataset and complementing it with the real dataset. Still, the high amount of images from the *Unity* dataset gets the best mean results using only the RGB channels, this makes evident that the quality from the NIR data from this dataset is considerably low, but not negligible for improvement.

Figure 7 and Figure 9 show the IoU results graph for sunflower and sugar beets respectively. Comparing these, it can be seen that the performance with the sunflower is generally higher and the *Reduced Augmented* performance comes really close to that of the *Real* one in the sugar beets case [9].

Only the mean accuracy is shown in Figure 8 because most of the variations are highly random for each class label as it can be seen in Table 3 since they were very high (between 97 and 99.99 percent). Thus, the IoU represent the most significant validation method while the accuracy becomes an standard verification that the trend is in fact consistent. Adding the NIR channel does increase the mean result and the best performance comes from using the *Reduced Augmented* dataset to train the CNN.

Table 4: Classification mean results (in percentage) for classification over 820 sugar beet images used for evaluation performance [9].

Method	Real	Unity	Unity + Real
Plant IoU RGB	83.35	60.83	73.51
Plant IoU RGBN	83.07	71.20	75.61
Weed IoU RGB	50.49	22.54	31.57
Weed IoU RGBN	54.09	29.57	36.82
Ground IoU RGB	98.56	98.14	98.41
Ground IoU RGBN	98.53	97.97	98.73
mean IoU RGB	77.47	60.50	67.83
mean IoU RGBN	78.56	66.25	70.38
Plant Accuracy RGB	99.04	97.23	98.63
Plant Accuracy RGBN	99.18	98.33	98.70
Weed Accuracy RGB	98.93	97.75	98.59
Weed Accuracy RGBN	99.03	98.18	98.70
Ground Accuracy RGB	99.99	99.97	99.99
Ground Accuracy RGBN	99.99	99.97	99.99
mean Accuracy RGB	99.32	98.18	99.07
mean Accuracy RGBN	99.40	98.82	99.16

7 Conclusion

The addition of the NIR channel to the training datasets for the selected CNN did provide a significant improvement for the sunflower-weed segmentation classification for both the intersection over union and the accuracy evaluation methods. This validates the addition of the NIR channel as clear solution to increase performance for plant-weed classification segmentation for sunflowers for both real images and synthetic images, including cases were the original dataset has both qualitative and quantitative limitations.

It is clear that any performance improvement comes directly from the quality of the images in the dataset, moreover from the quality of the data in each channel. The mean IoU performance increased considerably when hand-picked images were selected to build a smaller dataset which included only good quality images from the synthetic dataset plus the real images available, obtaining the best result. This provided a mean IoU result of 78.98% which is even higher than the best mean IoU result from our previous research of 78.56% using a real dataset of 1034 real images. Another important factor is that the amount of images for classification evaluation was higher in our previous research for sugar beets, 820 real images compared to 105 real images in our sunflower case.

The obtained results show that a dataset with a high amount of images could potentially add noise to the CNN deep learning training which provides a sub-optimal overall performance and it is advisable to consider creating subsets selecting the best quality images by hand, especially for images that are synthetically created with high levels of randomness.

This concludes part of our future work established in our previous paper where we mentioned we would add more plants to the simulator. It was possible to reuse the 3D mesh we previously had for the sugar beets and just replace the textures but the leveled structure of the sunflower plant made it a requirement that the procedural code followed certain pattern for the leaves to build realistic sunflowers for image generation.

References

1. The future of food and agriculture: trends and challenges, OCLC: ocn979567879
2. Migration, agriculture and rural development p. 20
3. GIMP (Sep 2020), https://www.gimp.org, [Online; accessed 1. Oct. 2020]
4. Unreal Engine | The most powerful real-time 3D creation platform (Sep 2020), https://www.unrealengine.com/en-US, [Online; accessed 30. Sep. 2020]
5. Abeysirigoonawardena, Y., Shkurti, F., Dudek, G.: Generating adversarial driving scenarios in high-fidelity simulators. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 8271–8277 (2019)
6. Acker, O.V., Lachish, O., Burnett, G.: Cellular automata simulation on FPGA for training neural networks with virtual world imagery. vol. abs/1711.07951 (2017), http://arxiv.org/abs/1711.07951
7. Ashtari, N., Bunt, A., McGrenere, J., Nebeling, M., Chilana, P.K.: Creating augmented and virtual reality applications: Current practices, challenges, and opportunities. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. pp. 1–13. ACM. https://doi.org/10.1145/3313831.3376722, https://dl.acm.org/doi/10.1145/3313831.3376722
8. Bah, M., Hafiane, A., Canals, R.: Deep learning with unsupervised data labeling for weed detection in line crops in UAV images. Remote Sensing **10**(11), 1690 (Oct 2018). https://doi.org/10.3390/rs10111690, https://doi.org/10.3390/rs10111690
9. Carbone, C.S., Potena., C., Nardi., D.: Simulation of near infrared sensor in unity for plant-weed segmentation classification. In: Proceedings of the 10th International Conference on Simulation and Modeling Methodologies, Technologies and Applications - Volume 1: SIMULTECH,. pp. 81–90. INSTICC, SciTePress (2020). https://doi.org/10.5220/0009827900810090
10. Carbone, C., Garibaldi, O., Kurt, Z.: Swarm robotics as a solution to crops inspection for precision agriculture **3**(1), 552. https://doi.org/10.18502/keg.v3i1.1459, https://knepublishing.com/index.php/KnE-Engineering/article/view/1459
11. Carbone, C., Garibaldi, O., Kurt, Z.: Swarm robotics as a solution to crops inspection for precision agriculture. KnE Engineering **3**(1), 552 (Feb 2018). https://doi.org/10.18502/keg.v3i1.1459, https://doi.org/10.18502/keg.v3i1.1459
12. Chebrolu, N., Lottes, P., Schaefer, A., Winterhalter, W., Burgard, W., Stachniss, C.: Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields. The International Journal of Robotics Research **36**(10), 1045–1052 (2017). https://doi.org/10.1177/0278364917720510, https://doi.org/10.1177/0278364917720510
13. Di Cicco, M., Potena, C., Grisetti, G., Pretto, A.: Automatic model based dataset generation for fast and accurate crop and weeds detection. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5188–5195. IEEE. https://doi.org/10.1109/IROS.2017.8206408, http://ieeexplore.ieee.org/document/8206408/

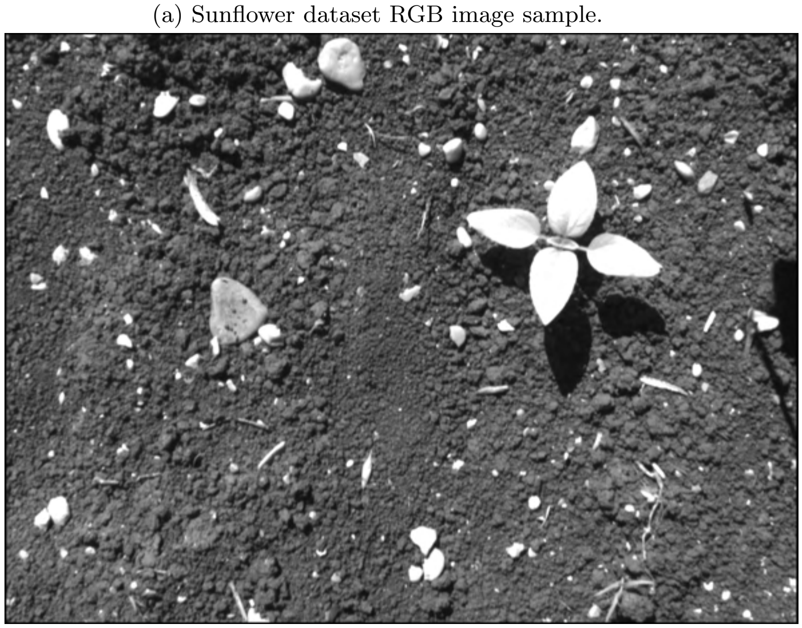
14. Duhan, J.S., Kumar, R., Kumar, N., Kaur, P., Nehra, K., Duhan, S.: Nanotechnology: The new perspective in precision agriculture. Biotechnology Reports **15**, 11–23 (Sep 2017). https://doi.org/10.1016/j.btre.2017.03.002, https://doi.org/10.1016/j.btre.2017.03.002
15. Fawakherji, M., Youssef, A., Bloisi, D., Pretto, A., Nardi, D.: Crop and weeds classification for precision agriculture using context-independent pixel-wise segmentation. In: 2019 Third IEEE International Conference on Robotic Computing (IRC). IEEE (Feb 2019). https://doi.org/10.1109/irc.2019.00029, https://doi.org/10.1109/irc.2019.00029
16. G, S., S, V., S, S., Suganya, G.: Application of neuroevolution in autonomous cars (2020)
17. Hossain, S., , Lee, D.J.: Autonomous-driving vehicle learning environments using unity real-time engine and end-to-end CNN approach. Journal of Korea Robotics Society **14**(2), 122–130 (May 2019). https://doi.org/10.7746/jkros.2019.14.2.122, https://doi.org/10.7746/jkros.2019.14.2.122
18. Hossain, S., Fayjie, A., Doukhi, O., Lee, D.: CAIAS Simulator: Self-driving Vehicle Simulator for AI Research, pp. 187–195 (01 2019). https://doi.org/10.1007/978-3-030-00979-$3_1$9
19. ISPA: Home | international society of precision agriculture (2020), https://www.ispag.org/
20. Justesen, N., Bontrager, P., Togelius, J., Risi, S.: Deep learning for video game playing. IEEE Transactions on Games **12**(1), 1–20 (2020)
21. Kaviratna, N.: Testing self driving cars with game development tools. In: Special Interest Group on Computer Graphics and Interactive Techniques Conference Talks. ACM (Aug 2020). https://doi.org/10.1145/3388767.3408334, https://doi.org/10.1145/3388767.3408334
22. Khanal, S., Fulton, J., Shearer, S.: An overview of current and potential applications of thermal remote sensing in precision agriculture. Computers and Electronics in Agriculture **139**, 22–32 (Jun 2017). https://doi.org/10.1016/j.compag.2017.05.001, https://doi.org/10.1016/j.compag.2017.05.001
23. Kim, T.S., Peven, M., Qiu, W., Yuille, A., Hager, G.D.: Synthesizing attributes with unreal engine for fine-grained activity analysis. In: 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW). pp. 35–37 (Jan 2019). https://doi.org/10.1109/WACVW.2019.00013
24. Kim, T.S., Peven, M., Qiu, W., Yuille, A., Hager, G.D.: Synthesizing attributes with unreal engine for fine-grained activity analysis. In: 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW). pp. 35–37 (2019)
25. Kolodny, L.: Fixed-wing drones not quite taking off in commercial market, a new DroneDeploy study finds \textbar TechCrunch. https://techcrunch.com/2016/08/15/fixed-wing-drones-not-quite-taking-off-in-commercial-market-a-new-dronedeploy-study-finds/ (2017), visited on 2017-08-09
26. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Communications of the ACM **60**(6), 84–90 (May 2017). https://doi.org/10.1145/3065386, https://doi.org/10.1145/3065386
27. Li, N., Zhang, X., Zhang, C., Guo, H., Sun, Z., Wu, X.: Real-time crop recognition in transplanted fields with prominent weed growth: A visual-attention-based approach. IEEE Access **7**, 185310–185321 (2019). https://doi.org/10.1109/access.2019.2942158, https://doi.org/10.1109/access.2019.2942158
28. Liu, O., Rakita, D., Mutlu, B., Gleicher, M.: Understanding human-robot interaction in virtual reality. In: 2017 26th IEEE International

Symposium on Robot and Human Interactive Communication (RO-MAN). pp. 751–757. IEEE. https://doi.org/10.1109/ROMAN.2017.8172387, http://ieeexplore.ieee.org/document/8172387/
29. Lottes, P., Behley, J., Milioto, A., Stachniss, C.: Fully convolutional networks with sequential information for robust crop and weed detection in precision farming. IEEE Robotics and Automation Letters **3**(4), 2870–2877 (Oct 2018). https://doi.org/10.1109/lra.2018.2846289, https://doi.org/10.1109/lra.2018.2846289
30. Lottes, P., Stachniss, C.: Semi-supervised online visual crop and weed classification in precision farming exploiting plant arrangement. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE (Sep 2017). https://doi.org/10.1109/iros.2017.8206403, https://doi.org/10.1109/iros.2017.8206403
31. Martinez, M., Sitawarin, C., Finch, K., Meincke, L., Yablonski, A., Kornhauser, A.: Beyond grand theft auto v for training, testing and enhancing deep learning in self driving cars (2017)
32. Matuszewski, J., Pietrow, D.: Deep learning neural networks with using pseudo-random 3d patterns generator. In: 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S T). pp. 262–267 (2019)
33. Milioto, A., Stachniss, C.: Bonnet: An Open-Source Training and Deployment Framework for Semantic Segmentation in Robotics using CNNs. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2019)
34. Milioto, A., Lottes, P., Stachniss, C.: Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns. CoRR **abs/1709.06764** (2017), http://arxiv.org/abs/1709.06764
35. Nilsson, R.: Battery Performance Comparison Of Unreal Engine 4 And Unity Applications Running On Android. , Department of Software Engineering, backup Publisher: Blekinge Institute of Technology, Department of Software Engineering
36. Patrício, D.I., Rieder, R.: Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. Computers and Electronics in Agriculture **153**, 69–81 (Oct 2018). https://doi.org/10.1016/j.compag.2018.08.001, https://doi.org/10.1016/j.compag.2018.08.001
37. Phisannupawong, T., Kamsing, P., Torteeka, P., Channumsin, S., Sawangwit, U., Hematulin, W., Jarawan, T., Somjit, T., Yooyen, S., Delahaye, D., Boonsrimuang, P.: Vision-based spacecraft pose estimation via a deep convolutional neural network for noncooperative docking operations. Aerospace **7**(9), 126 (Aug 2020). https://doi.org/10.3390/aerospace7090126, https://doi.org/10.3390/aerospace7090126
38. Potena, C., Nardi, D., Pretto, A.: Fast and accurate crop and weed identification with summarized train sets for precision agriculture. In: Intelligent Autonomous Systems 14, pp. 105–121. Springer International Publishing (2017). https://doi.org/10.1007/978-3-319-48036-7$_9$, $https://doi.org/10.1007/978-3-319-48036-7_9$
39. Richter, S.R., Vineet, V., Roth, S., Koltun, V.: Playing for data: Ground truth from computer games. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision – ECCV 2016. pp. 102–118. Springer International Publishing, Cham (2016)
40. Sa, I., Chen, Z., Popovic, M., Khanna, R., Liebisch, F., Nieto, J., Siegwart, R.: weedNet: Dense semantic weed classification using multispectral images and MAV for smart farming. IEEE Robotics and Automation Letters **3**(1), 588–595 (Jan 2018). https://doi.org/10.1109/lra.2017.2774979, https://doi.org/10.1109/lra.2017.2774979
41. Sa, I., Popović, M., Khanna, R., Chen, Z., Lottes, P., Liebisch, F., Nieto, J., Stachniss, C., Walter, A., Siegwart, R.: WeedMap: A large-scale semantic weed mapping framework using aerial multispectral imaging and deep neural network for precision

farming. Remote Sensing **10**(9), 1423 (Sep 2018). https://doi.org/10.3390/rs10091423, https://doi.org/10.3390/rs10091423

42. Stafford, J.V.: Implementing precision agriculture in the 21st century. Journal of Agricultural Engineering Research **76**(3), 267–275 (Jul 2000). https://doi.org/10.1006/jaer.2000.0577, https://doi.org/10.1006/jaer.2000.0577
43. Technologies: Unity - Unity (Sep 2020), https://unity.com, [Online; accessed 30. Sep. 2020]
44. Torrado, R.R., Bontrager, P., Togelius, J., Liu, J., Perez-Liebana, D.: Deep reinforcement learning for general video game ai. In: 2018 IEEE Conference on Computational Intelligence and Games (CIG). pp. 1–8 (2018)
45. Yang, C., Lee, T., Huang, C., Hsu, K.: Unity 3d production and environmental perception vehicle simulation platform. In: 2016 International Conference on Advanced Materials for Science and Engineering (ICAMSE). pp. 452–455 (Nov 2016). https://doi.org/10.1109/ICAMSE.2016.7840349

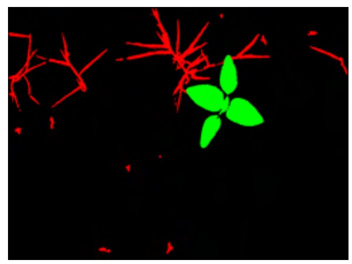
(a) Sunflower dataset RGB image sample.



(b) Sunflower dataset NIR image sample.

Fig. 1: Images from the sunflower real dataset. In the NIR image (below) it can be seen that the brightness makes some of the rocks on the image bottom look almost fully white as one of the sunflower leaves.

(a) RGB.



(b) Labeled.

(c) Cropped.

Fig. 2: Weed texture extracted from sunflower dataset, original RGB on the left, labeled version on the right, Green labels are the sunflower plants and red labels are the weeds. Bottom shows texture showing only the weeds cropped.

(a) Sunflower RGB.

(b) Sunflower NIR.

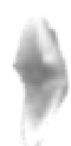
(c) Root leaf RGB.

(d) Root leaf NIR.

(e) Head leaf RGB.

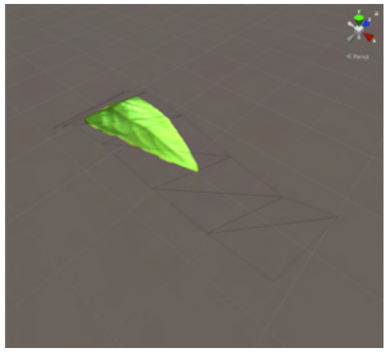
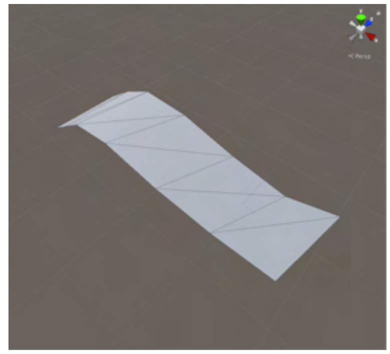
(f) Head leaf NIR.

(g) Center leaf RGB.



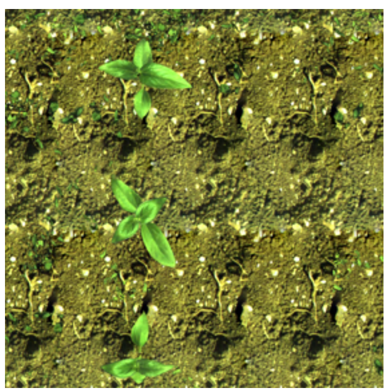
(h) Center leaf NIR.

Fig. 3: Texture extraction from dataset image for Unity crop simulation.

(a) 3D leaf with shaders. (b) 3D leaf mesh.

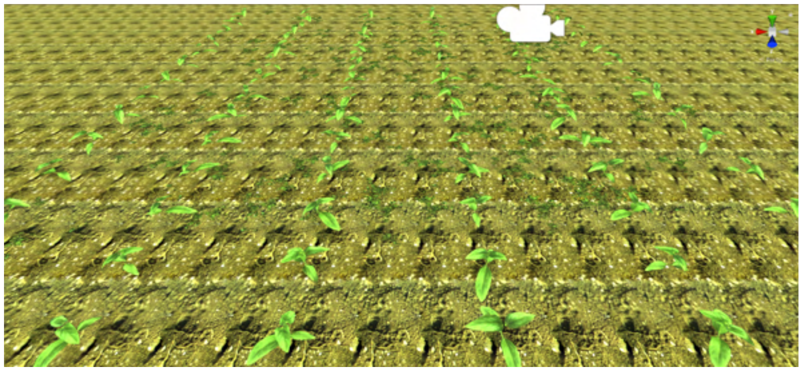
Fig. 4: 3D leaf mesh used to procedurally build sunflower plants.



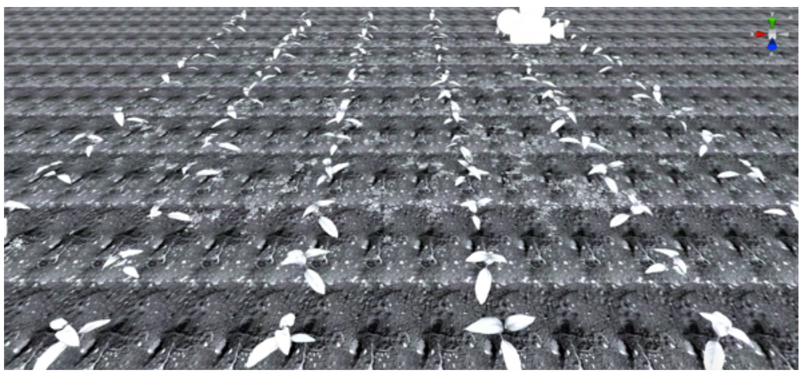

(a) Unity RGB. (b) Unity NIR.


(c) Real RGB. (d) Real NIR.

Fig. 5: Comparison between synthetic sunflower in unity (top) and a real one (bottom), including RGB (left) and NIR (right) comparison.



(a) Unity RGB field.



(b) Unity NIR field.

(c) Unity labeled masks field.

Fig. 6: Sunflower field in Unity: RGB (top), NIR (middle) and labeled masks (bottom).

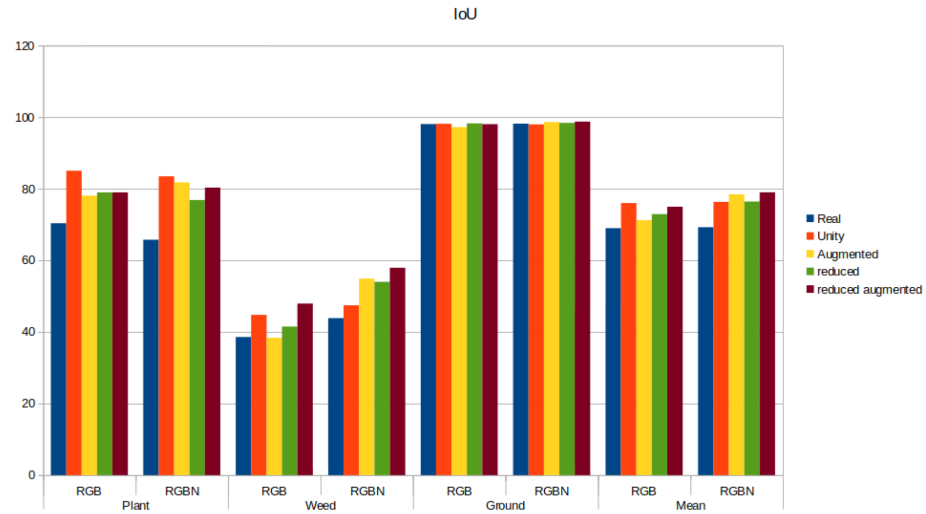


Fig. 7: Sunflower intersection over union classification results from Table 3 using Bonnet trained with different datasets from Table 1.

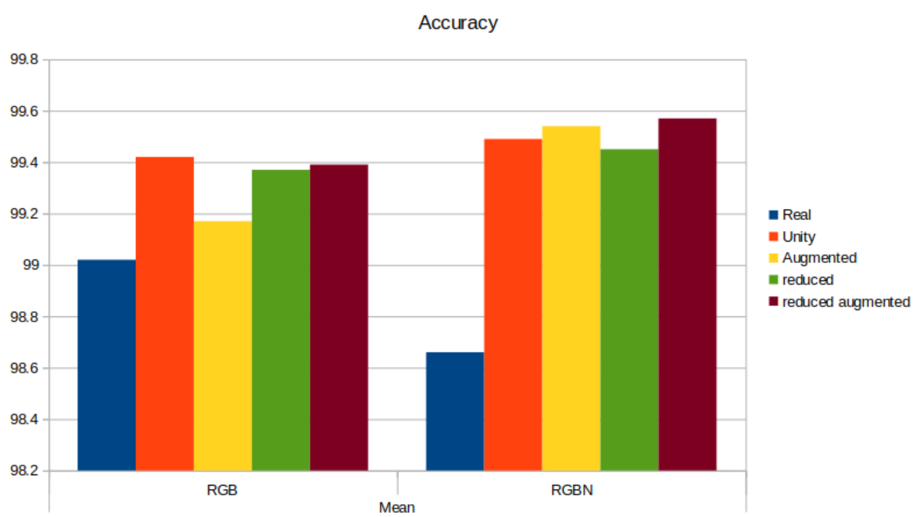


Fig. 8: Sunflower accuracy classification mean results from Table 3 using Bonnet trained with different datasets from Table 1.

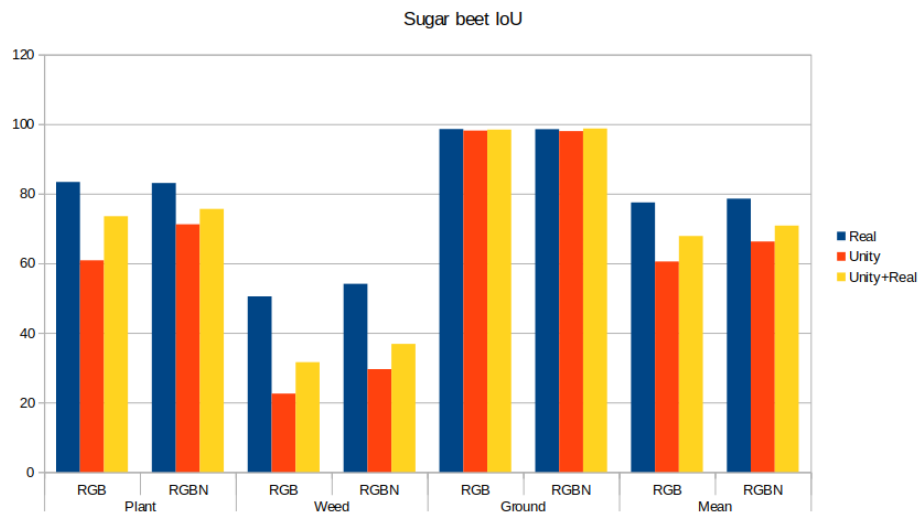


Fig. 9: Sugar beets intersection over union classification results from Table 4 using Bonnet CNN trained with different datasets from Table 2 [9].