This paper has been accepted for publication in Robotics and Autonomous Systems
(DOI: 10.1016/j.robot.2021.103861)

Multi-Spectral Image Synthesis for Crop/Weed Segmentation in Precision Farming

Mulham Fawakherji[1], Ciro Potena[2], Alberto Pretto[3],
Domenico D. Bloisi[4], and Daniele Nardi[1]

[1] Department of Computer, Control, and Management Engineering,
Sapienza University of Rome, Rome, Italy
{fawakherji,nardi}@diag.uniroma1.it
[2] Engineering Department, Roma Tre University, Rome, Italy
cpotena@os.uniroma3.it
[3] Department of Information Engineering, University of Padua, Padua, Italy
alberto.pretto@dei.unipd.it
[4] Department of Mathematics, Computer Science, and Economics
University of Basilicata, Potenza, Italy
domenico.bloisi@unibas.it

Abstract. An effective perception system is a fundamental component for farming robots, as it enables them to properly perceive the surrounding environment and to carry out targeted operations. The most recent methods make use of state-of-the-art machine learning techniques to learn a valid model for the target task. However, those techniques need a large amount of labeled data for training. A recent approach to deal with this issue is data augmentation through Generative Adversarial Networks (GANs), where entire synthetic scenes are added to the training data, thus enlarging and diversifying their informative content. In this work, we propose an alternative solution with respect to the common data augmentation methods, applying it to the fundamental problem of crop/weed segmentation in precision farming. Starting from real images, we create semi-artificial samples by replacing the most relevant object classes (i.e., crop and weeds) with their synthesized counterparts. To do that, we employ a conditional GAN (cGAN), where the generative model is trained by conditioning the shape of the generated object. Moreover, in addition to RGB data, we take into account also near-infrared (NIR) information, generating four channel multi-spectral synthetic images. Quantitative experiments, carried out on three publicly available datasets, show that (i) our model is capable of generating realistic multi-spectral images of plants and (ii) the usage of such synthetic images in the training process improves the segmentation performance of state-of-the-art semantic segmentation convolutional networks.

1 Introduction

Precision agriculture is a farming management concept based on observing, measuring, and responding to inter and intra-field variability in crops [12]. A key

Please cite this paper as: M. Fawakherji, C. Potena,
A. Pretto, D.D. Bloisi, D. Nardi
Multi-Spectral Image Synthesis for Crop/Weed Segmentation in Precision Farming,
Robotics and Autonomous Systems, Volume 146, 2021
https://doi.org/10.1016/j.robot.2021.103861

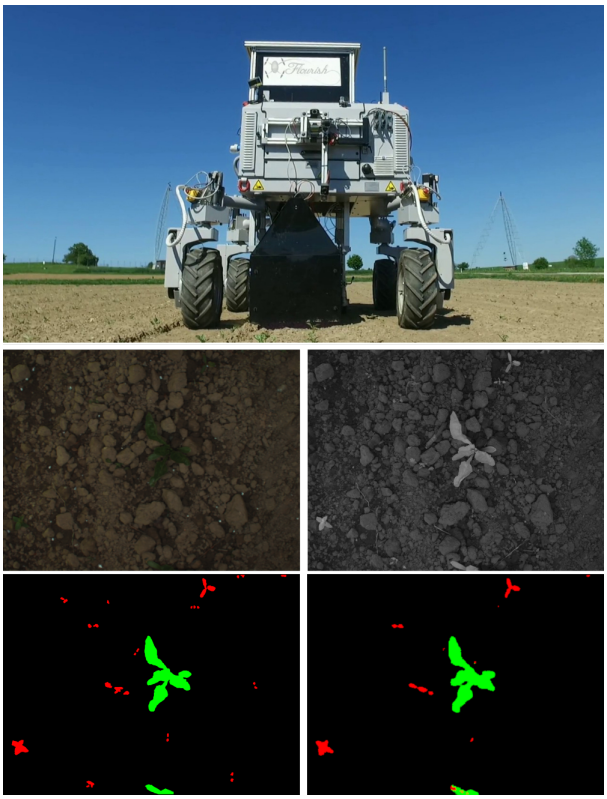


Fig. 1: Top row: The BOSCH Bonirob farming robot used to collect the datasets considered in the experiments. The Bonirob is equipped with a Weed Intervention Module (the black structure between the wheels in the picture). This module consists of a perception system for weed classification and a multi-modal actuation systems for weeds removal. Middle row: From left to right, synthetic RGB and synthetic NIR samples, respectively. Bottom row: From left to right, the pixel-wise ground truth and the result obtained by using a semantic segmentation deep neural network, respectively.

objective in precision agriculture is the minimization of environmental impacts by reducing the reliance on chemicals products such as herbicides or pesticides. Farming robots (e.g., see Fig. 1, top row) can play an important role in this mission, as they can perform precise weed control through selective treatment applications (e.g., [25]).

A fundamental requirement to perform selective treatments through robots is to build an effective perception module capable of identifying and localizing crop and weeds in the field and thus trigger the proper weeding actions. The most commonly adopted approaches use image processing to tackle this problem and rely on machine learning methods, such as Convolutional Neural

Networks (CNNs) [16, 24, 29]. These data-driven methods allow to train powerful visual classifiers that report high classification performance. However, their performance strongly depends on the size and variety of the training dataset [35]. This problem is well-known and has been addressed in many different ways (see, among others, [4, 24]). More recent approaches address this problem by leveraging Generative Adversarial Networks (GANs) [7, 30]. These methods allow to train, in an unsupervised manner, powerful generative models capable of synthesizing photo-realistic images that can be used to increase and diversify the original training datasets. This results in an improved generalization capability of the learned visual classifiers.

In this work, we address the crop and weeds detection task in terms of a semantic image segmentation system capable of identifying the crop in real images at pixel level, distinguishing it from weeds. Such semantic segmentation task can be effectively tackled by using state-of-the-art data driven, deep learning-based methods such as [20, 27]. The main disadvantage of such approaches is the need to access large amounts of data provided with accurate pixel-wise semantic annotations (e.g., see the bottom left picture in Fig. 1). A way to automatically generate at least part of this data is desirable. In this context, we propose a novel methodology to synthesize photo-realistic images by using a generative adversarial method. Unlike the conventional uses of GANs, which aims to train a model to generate an entire scene, we generate semi-artificial images by replacing only the regions of the scene corresponding to the objects that are relevant to the target perception task (crop and weed plants in our case) with synthesized, photo-realistic counterparts. The intuition behind this idea is that, usually, vision-based learned classifiers are not able to equally generalize across all the target classes, which in turn can lead to unbalanced classification performance. To achieve our goal, we use a conditional GAN (cGAN), where the generative model is trained by conditioning the shape of the generated object. This allows to synthesize new realistic objects while keeping their original footprint onto the image, since the generative model receives as an input constraint the object shapes extracted from real objects.

The main contributions of this work are three-fold. First, we use a cGAN to learn only the data distribution associated to a subset of the target classes, allowing to train more compact generative models and to create photo-realistic training samples in a faster and more effective way. Second, we perform a quantitative study on our cGAN that estimates the amount of real data needed to generate consistent results. Third, we use NIR information in order to generate four channel multi-spectral synthetic images.

Using the NIR channel helps to improve accuracy in activities that require vegetation detection. Due to the photosynthesis, healthy green plants absorb more solar energy in the visible spectrum, causing a low reflectance level in the RGB channels. Similarly, the reflectance of the near-infrared spectrum is affected by the same phenomena with opposite results, with a high reflectance level in the NIR channel, where generally 10% or less of radiation is absorbed [33].

As a further contribution, we created and made publicly available with this paper a new pixel-wise labeled dataset, the *Sunflower Dataset*, which contains a large number of multi-spectral annotated images acquired over different growing stages in a sunflowers field. The pixel-wise labels highlight the three classes: crop, weed and soil. The Sunflower Dataset and the project's code are available at:
`https://bit.ly/3hHenpE`

To evaluate the effectiveness of the proposed architecture, we report experiments on three publicly available farming datasets, showing that our model is capable of generating realistic 512×512 multi-spectral images of plants (see for instance the middle row of Fig. 1), and that the usage of these synthetic images during the training process improves the segmentation performance of state-of-the-art semantic segmentation deep neural networks (SSNs).

The remainder of the paper is organized as follows. After discussing related work in Section 2, a brief description of GANs and cGANs is given in Section 3. The proposed method is presented in Section 4, while experimental results are shown in Section 5. Finally, conclusions are drawn in Section 6.

2 Related Work

A robust crop/weed classification module is an essential component for autonomous farming robots, as it enables the platform to properly perceive the environment and to carry out an efficient weed control policy. The problem has been extensively investigated over the last years and the proposed approaches can be roughly split in two main categories:

1. Classifiers based on hand-crafted features.
2. Classifiers based on learned features.

The methods of the first group usually have limited generalization capabilities, depending on the choice of the features to process. The approaches in the second category have better generalization capabilities, at the cost of annotating large datasets of images, which is a tedious and time-consuming process. In this section, we focus on crop/weed approaches belonging to the two categories mentioned above. Moreover, we provide a discussion about methods that address the dataset annotation issue.

2.1 Classifiers Based on Hand-crafted Features

Methods in this class aim at finding a suitable set of features that have good discrimination properties among the target plant classes. Haug *et al.* [11] propose a plant classification method that is capable of distinguishing carrots and weeds by using RGB and NIR images. The reported accuracy is around 93.8%. Lottes *et al.* [16] propose a sugar beets and weeds classification system based on a multi-spectral camera mounted on the robot. The method performs, in sequence, a vegetation detection, an object-based features extraction, a random forest classification, and a smoothing post-process through a Markov random

field. Experiments have been carried out in different sugar beets fields reporting good classification performance. This method has been extended in [15], where the crop/weed classification data are acquired using a camera mounted on a light-weight UAV. The system has been tested in two farms located in Germany and Switzerland, showing good generalization properties and the ability to classify individual plants. Despite the positive results, the methods based on hand-crafted features are strictly dependent on the choice of the features, which limits their generalization capabilities.

2.2 Classifiers Based on Learned Features

Machine learning methods, and more specifically CNNs, offer the potential to overcome the inflexibility of handcrafted vision pipelines, by allowing to develop effective end-to-end classification methods. In this regards, CNNs are usually applied in a pixel-wise fashion, operating on image patches, provided by a sliding window approach. Following this idea, Potena *et al.* [24] propose a crop/weed classification architecture composed of a cascade of CNNs. The first CNN detects the vegetation, which is successively used as the input for a second, deeper, CNN that classifies vegetation pixels into crop or weeds. McCool *et al.* [18] propose a three stage approach. They start from a pre-trained CNN model with state-of-the-art performance, but a high computational cost. Then, a model compression is performed, leading to a faster CNN. Finally, they combine several lightweight models into a mixture model to enhance the performance. They report an accuracy around 93.9%. Mortensen *et al.* [22] use a CNN to estimate the in-field biomass and crop composition. Their method is a modified version of the well-known VGG-16 deep neural network. The reported accuracy is 79% with seven classes of objects.

Differently from classification CNNs, semantic segmentation deep neural networks (SSNs) take images of arbitrary size as input and produce segmented output of corresponding size, without relying on local patches.

Among the many SSNs proposed in the literature, one of the most commonly adopted in crop/weed segmentation is SegNet [1], which is a deep encoder-decoder architecture for multi-class pixelwise segmentation. Di Cicco *et al.* [4] trained SegNet with real and synthetic images reporting good segmentation performance. Sa *et al.* [29] use SegNet for dense semantic weed classification with multispectral images collected by a Micro Aerial Vehicle (MAV). A similar encoder-decoder architecture is exploited by Milioto *et al.* [19]. They augment the RGB input image with task-relevant background knowledge to speed up the training and to better generalize to new crop fields. We exploit a similar idea in our previous work [6], where we propose a pipeline with multiple data channels to support the input of a CNN by using more vegetation indices. These additional information aid the CNN to achieve a good generalization to different crop types. Lottes *et al.* [14] propose a crop/weed classification system that, in addition to a Fully Convolutional Network (FCN) [13], also exploits the crop arrangement information that is observable from the image sequences. This in-

creases the segmentation performance and the generalization capabilities of the net to previously unseen fields under varying environmental conditions.

2.3 Labeling Effort Reduction

The major drawback of both CNNs and SSNs based architectures is that the level of expressiveness is limited by the size of the training dataset. In the context of precision farming, collecting large annotated datasets involves a significant effort. Datasets should be acquired across different growth stages and weather conditions. Moreover, in the case of SSNs, the pixel-wise annotation process is tedious and extremely time consuming. As a matter of fact, the size of pixel-wise annotated datasets is usually relatively small [35].

To cope with the above discussed problems, different solutions have been proposed in the literature. Potena *et al.* [24] propose a novel dataset summarization technique. The main idea is to condense an original, unlabeled, dataset by taking only the most informative images. The summarized dataset will thus lead to a reduced labeling effort while keeping a sufficient segmentation performance. Di Cicco *et al.* [4] use a state-of-the-art graphic engine to generate synthetic and realistic farming scenes. The generated scene, together with the corresponding ground truth data, are used to train the final CNN or to supplement an existing real dataset. Milioto *et al.* [19] propose a CNN that requires little data to adapt to the new, unseen environment. The reported results show a segmentation accuracy around 96% and a fast re-adaptation to the new environments. Sa *et al.* [29] deal with the labeling effort by exploiting different fields with varying herbicide levels, resulting in field patches containing only either crop or weed. This enables to exploit a simple vegetation index as a feature for automatic ground truth generation. Although the methods described so far can successfully reduce the annotation effort, they may not yet achieve the segmentation performance of a fully trained SSN.

More recent approaches make use of GANs. Giuffrida *et al.* [7] exploit a conditional GAN to generate 128×128 synthetic *Arabidopsis* plants, with the possibility to decide the desired number of leaves for the final plant. Their method has been tested using a leaf counting algorithm in order to show how the addition of synthetic data helps to avoid overfitting and to improve the accuracy. Madsen *et al.* [17] leverage a GAN to generate artificial image samples of plant seedlings to mitigate for the lack of training data. Their method is capable of generating nine distinct plant species, while increasing the overall recognition accuracy.

As a difference with respect to the last discussed methods, in this work we propose to generate *multi-spectral* views of agricultural scenes by synthesizing only the objects that are relevant for semantic segmentation purposes. Our method starts by generating a synthetic plant using a cGAN, then the real plant in the image is replaced by a synthetic generated one in order to create a new, semi-artificial image.

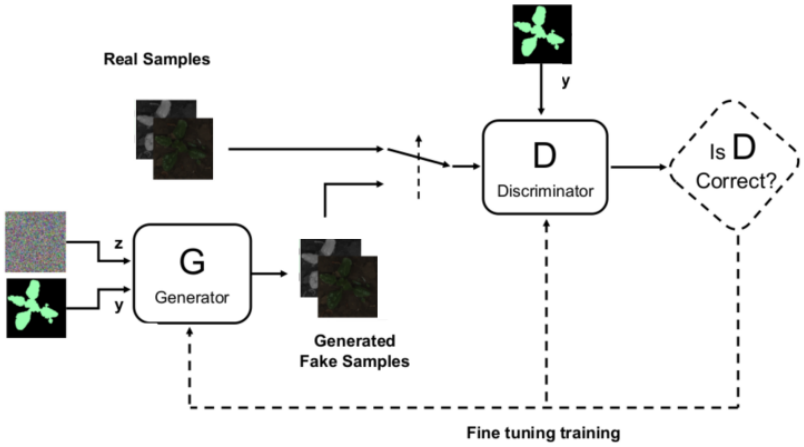


Fig. 2: The cGAN generator learns a nonlinear function G that maps an input mask to a photo-realistic image. The cGAN discriminator learns a function D that discerns real from synthesized images produced by G.

3 Preliminaries

3.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) [8] can estimate generative models through an adversarial training, which simultaneously trains two networks:

1. A generative model G, whose objective is to capture a data distribution.
2. A discriminative model D that outputs a single scalar. The goal for D is to estimate the probability that a sample is actually a real data and not a sample synthetically generated by G.

Given any data distribution $p_{data}(x)$ and a prior input noise distribution $p_z(z)$ (which is typically uniform or Gaussian), the mapping to the data space is represented by $G(z)$, where G is the generative model with its distribution p_g. Let us also define the discriminator D as a function that outputs a single scalar $D(x)$ representing the probability that x comes from the real data space rather than from p_g.

The training process is carried out by maximizing the probability of D to assign a correct label to the generated and to the real samples, while G is trained to learn the distribution p_g over the data space x so that D can hardly assign them the correct label. G and D are trained in an unsupervised manner by a two-player min-max game that is given by:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim p_z(z)}[log(1-D(G(z)))] \quad (1)$$

where $\mathbb{E}$ and log are the expectation and logarithmic operators, respectively. D and G are trained simultaneously until they cannot both improve because

$p_g = p_{data}$ and the discriminator is unable to distinguish between the two distributions, i.e., $D(x) = 1/2$.

3.2 Conditional Generative Adversarial Networks

Conditional GANs (cGANs) [21] extend the GAN concept by conditioning both D and G through extra data y. The cGAN scheme is shown in Fig. 2. It is worth noticing that y can represent any kind of auxiliary information (in [7], for instance, y represents the number of leaves of the synthesized plant) and it is fed into both the generator G and discriminator D as an additional input layer. In the cGAN scheme, G attempts to synthesize realistic images (i.e., fake samples) from the y domain, while D receives samples from both x and y domains and attempts to discern between $(real, real)$ and $(fake, real)$ image pairs.

The loss function of a cGAN can be expressed as:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[logD(x,y)] + \mathbb{E}_{z\sim p_z(z)}[log(1-D(G(z,y)))] \quad (2)$$

4 Proposed Method

Our goal is to develop an algorithm capable of synthesizing realistic agricultural scenes. Let us define our data distribution $p_{data}(x)$ as a set of images collected by a moving robot in a cultivated field. The images are acquired by a multispectral camera that collects NIR images in addition to, and registered with, RGB images. The dataset is annotated in a pixel-wise manner and, for each image, we have a corresponding *total mask* containing information about crop, weed, and soil pixels.

To synthesize new realistic annotated images we need to accomplish two main tasks. The first task consists in extracting a *plant mask* from the total image mask. A plant mask is a binary image where the plant pixels that we want to learn are set to 1 and everything else is set to 0. The second task consists in learning a function $G : z, y \rightarrow x$ that maps the plant mask y in input to a realistic multispectral image. The mapping function G is implemented in a cGAN that contains an implicit model of the conditional probability distribution $p(x|y)$ learned by training. The resulting images will be used as data augmentation to train a deep learning model for crop/weed segmentation. Solving the tasks discussed above has two advantages: 1) it permits to enlarge the training dataset and 2) it allows to diversify the data, thus significantly improving the generalization of the learned models.

The usage of cGAN as data augmentation tool is not novel and it has been explored in different fields, ranging from medical images, anomaly detection, image classification, and even in the decoding of the position, orientation, and binary ID of markers [7,10,31]. The data augmentation problem is usually addressed by training a generative model capable to reproduce an entire scene, which requires deep models, a large amount of training data, and high computational power. However, a full scene generation is redundant for our crop/weed segmentation

task, where plants are represented by a small percentage of the whole image pixels and the majority of pixels belongs to the background (soil).

Moreover, since the accuracy of a SSN can often vary significantly across classes, in our scenario we can augment the number of training samples only for the classes with a lower classification accuracy. In fact, while the network is able to accurately detect the soil, it usually miss-classifies the pixels belonging to crop and weeds (due to their similar visual appearance). In such a case, there is no need to increase the number of soil samples, while increasing the crop samples can provide a significant information gain.

In this work, rather than training a generator G capable to synthesize an entire scene, our idea is to focus on generating instances of some specific object classes, specifically the ones with the lowest segmentation accuracy. In the rest of this section, we describe the three major steps involved in the generation of realistic agricultural samples:

1. The training of the cGAN for learning the generative model.
2. The quantitative evaluation of the cGAN training results.
3. The composition of the synthetic farming scenes.

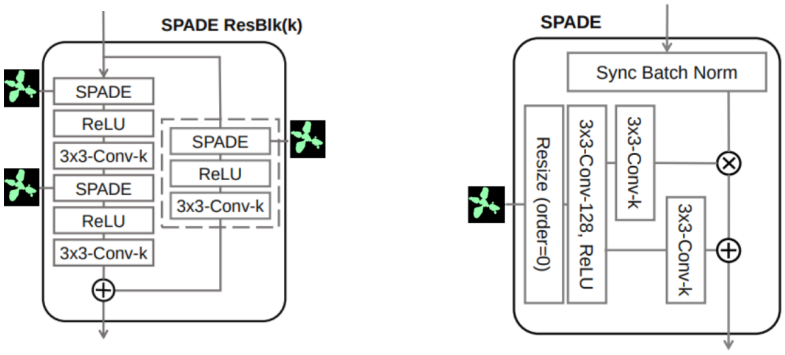
4.1 cGAN Architecture

The first step of our approach concerns the generation of photo-realistic images of specific classes of objects that populate an agricultural scene. We employ here the SPatially-Adaptive DEnormalization (SPADE) cGAN architecture [23]. Differently from other common cGANs, this type of network performs a semantic image synthesis by converting a semantic segmentation mask into a photo-realistic image. In other words, its input/output behavior is the opposite of an image segmentation network.

In the SPADE architecture, the image encoder encodes a real image into a latent representation for generating a mean and a variance vector. This architecture aims to sample from the learned model new realizations, modulating the style in terms of color and texture of the elements of interest, while unchanged keeping the original shapes used in conditioning. The generator in the SPADE architecture contains a series of basic components, called residual blocks (ResBlk). A SPADE ResBlk (shown in Fig. 3a) includes some SPADE elements (see Fig. 3b). The SPADE generator is built based on the pix2pixHD framework [34]. It starts with random noise in the input and uses the semantic map at every SPADE ResBlk layer. Using SPADE, it is also possible:

- to separate between semantic and style control;
- to change the final content, by modifying the semantic map;
- to change the style of the image, by modifying the random vector.

The SPADE discriminator takes in input a concatenation of the segmentation map with the original (or generated) image and decides if that image is real or fake. Also the discriminator architecture follows pix2pixHD and it uses a multi-scale design with instance normalization, with the difference that spectral



(a) SPADE ResBlk

(b) SPADE element

Fig. 3: SPADE Architecture. (a) SPADE ResBlk. (b) SPADE element. The term 3x3-Conv-k denotes a 3-by-3 convolutional layer with k convolutional filters. The segmentation map is resized to match the resolution of the corresponding feature map using nearest-neighbor downsampling.

normalization is applied to all the convolutional layers of the discriminator. The encoder is composed of six convolutional layers with stride 2 followed by two linear layers. It is responsible for producing the mean (μ) and covariance (σ^2).

Compared to its original version, we have made two major changes to the SPADE architecture:

1. The first change is in the SPADE image synthesis modalities. The original version takes as input RGB images and generates RGB images as well. In our case, to exploit the NIR channel, we enabled the network to work with four channel images and thus to generate multi-spectral images.
2. The second modification has been made to increase the size of the generated samples. The original version generates images with a resolution of 256×256, which may not be enough to generate all the possible object classes. Differently, we generate images with a resolution of 512×512.

4.2 cGAN Evaluation Metrics

Evaluating GANs is a very challenging task and several aspects need to be taken into consideration when defining metrics that can produce meaningful scores. These metrics should be capable to distinguish between generated and real samples, to detect overfitting, and to identify mode collapse and mode drop. Our goal is to check whether the learned generative model generalizes well with respect to the problem of photo-realistic crop/weed generation.

For most of the GANs presented in the literature, network inspection is qualitative only, based on a manual inspection to verify the fidelity of the generated sample. This kind of evaluation is still considered the best approach, but it is time-consuming, subjective, and often it can also be misleading.

In this paper, we employ an empirical evaluation based on quantitative metrics. The key idea is to use samples generated by the network and samples collected from the real dataset to extract features from both of them, and then to calculate performance using specific metrics. In particular, we employ six metrics: Inception Score, Mode Score, Kernel MMD, Wasserstein distance, Fréchet Inception Distande (FID) and 1-nearest neighbor (1-NN). For space constraints, we describe here only the Inception Score, being the most popular metric for evaluating GANs, while the definition of all other metrics can be found in [36].

Inception Score It is a metric capable of measuring not only the quality, but also the diversity of generated images using an external model, the Google Inception network [32], trained on the ImageNet dataset [28]. The Inception Score (IS) can be calculated using the following equation:

$$\mathrm{IS}(p_g) = e^{\mathbb{E}_{x\sim p_g}[KL(p_{\mathcal{M}}(l|\mathrm{x})\|p_{\mathcal{M}}(l))]} \tag{3}$$

By considering a pre-trained model $\mathcal{M}$, $p_{\mathcal{M}}(l \mid \mathrm{x})$ refers to the label distribution of x predicted by $\mathcal{M}$, and $p_{\mathcal{M}}(l) = \int_{\mathrm{x}} p_{\mathcal{M}}(l \mid \mathrm{x}) dp_g$, which gives the marginal of $p_{\mathcal{M}}(l \mid \mathrm{x})$ over the probability measure p_g. The expectation and the integral in $p_{\mathcal{M}}(l \mid \mathrm{x})$ can be approximated with independent and identically distributed samples from p_g. The KL operator represents the Kullback–Leibler divergence between the distributions $p_{\mathcal{M}}(l \mid \mathrm{x})$ and $p_{\mathcal{M}}(l)$.

We used the six metrics listed above as evaluation metrics to give a final quantitative intuition of how much the generated fake samples are close to the real data distribution. First, we identify a reference value by computing the metrics over two sets of samples from the real dataset. This process is repeated ten times with random selection at each time and getting the mean. Then, for each cGAN model output, we generate a set of fake samples and compute the metrics between the generated fake samples and the real data.

4.3 Agricultural Scene Composition

In the final step, our approach uses the crop and weeds RGB/NIR images described in Section 4.1 to create a realistic agricultural scene. To do so, we follow the scheme shown in Fig. 4.

First, we get the crop and weed masks from the total image mask. The mask is then resized to the SPADE network input size, which is 512×512 pixels for crop and 128×128 for weeds. This difference is due to the fact that weeds tend to be smaller than the crop. The SPADE network then generates a random, photo-realistic, crop/weeds instance by using the shape of the input mask and a random noise signal. The generated image is then replaced into the source image. Fig. 5 shows an example of the obtained synthetic image.

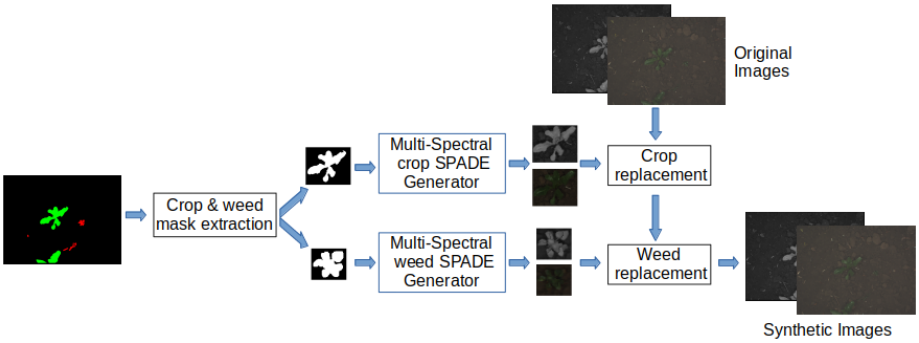


Fig. 4: Dataset creation for segmentation training. First the crop and weed masks are taken from the full mask, then new RGB+NIR crops and weeds are generated from these masks and pasted back into the original image.

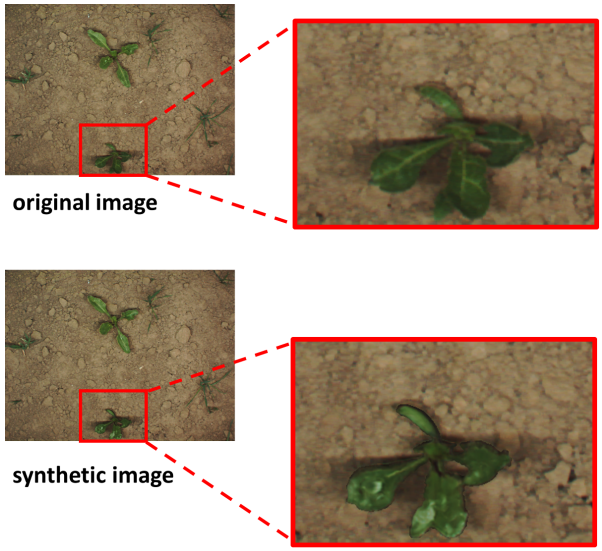


Fig. 5: An original and a synthetic image. The synthetic image is obtained by inserting in the original image a plant sample generated by using our cGAN.

5 Experimental Results

A quantitative evaluation has been carried out to show that by augmenting the training datasets with synthetic photo-realistic images generated with our model it is possible to:

1. Improve the generalization capability of the chosen segmentation network;
2. Increase the performance in crop/weed segmentation.

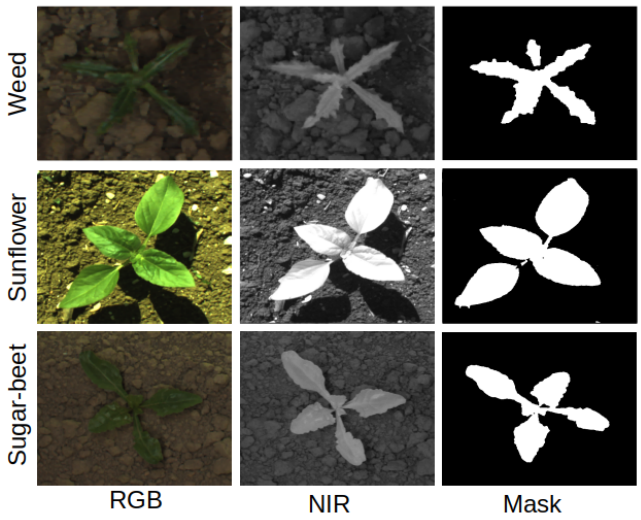


Fig. 6: Training data for our SPADE module. First row: examples of data used to train SPADE for weeds. Second row: examples of data used to train SPADE for sunflowers. Third row: examples of data used to train SPADE for sugar beets. The columns show (from left to right) RGB, NIR, and mask samples.

Moreover, the annotation effort is reduced, since the cGAN generates both the images and the masks. We also performed a quantitative study to test the amount of real data needed to train a SPADE cGAN model capable of generating good plant samples, i.e., synthetic images close enough to the real data.

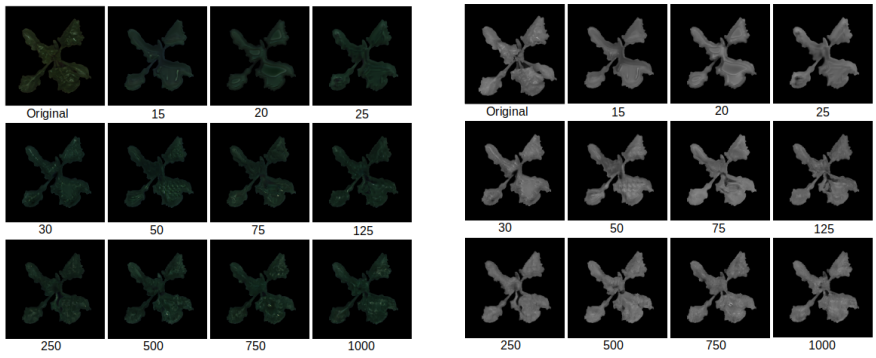
5.1 Experimental setup

We performed experiments on three publicly available datasets, considering two different types of crops, namely sugar beet and sunflower.

Sugar beet datasets. For sugar beet, we used two publicly available datasets: the Bonn dataset and the Stuttgart dataset [3]. Both datasets have been collected by using a BOSCH Bonirob farm robot moving on a sugar beet field across different weeks. The datasets are composed of images taken by a 1296×966 pixels 4-channels (RGB + NIR) JAI AD-13 camera, mounted on the robot and facing downward. An example of sugar beet taken from Bonn dataset is shown at the bottom of Fig. 6.

Sunflower dataset. In this work, we introduce a new dataset for crop/weed segmentation, called Sunflower dataset[5] that has been collected by the authors

[5] `http://www.diag.uniroma1.it/~labrococo/fsd/sunflowerdatasets.html`



(a) RGB (b) NIR

Fig. 7: Examples of SPADE models outputs. The number under each image represents the cardinality of the dataset used to train the cGAN.

of this paper. An example of sunflower taken from the Sunflower dataset is shown in the middle of Fig. 6. Data has been acquired by using a custom-built agricultural robot moving in a sunflower farm in Jesi, Italy. The dataset has been recorded in spring, across a period of one month, starting from the emergence stage of the crop plants, until the end of the useful period for the use of chemical treatments. As for the Bonn and the Stuttgart datasets, images were acquired using a 1296×966 pixels 4-channels (RGB + NIR) JAI AD-13 camera, mounted on the robot and facing downward. The Sunflower dataset, composed of 500 images, provides RGB and NIR images with pixel-wise annotation of 3 classes: crop, weed, and soil. It is organized into three subsets:

- *Jesi-05-12*, which includes images of sunflower crops in the emergence stage.
- *Jesi-05-18*, which includes images of sunflower crops in a subsequent growth stage.
- *Jesi-06-13*, which includes images of sunflower crops few days before the end of the period for using chemical treatments.

5.2 cGAN Impact Evaluation

Our method requires a certain number of labeled real images to effectively train the cGAN network. In this experiment, we measure the minimum number of images needed to train the cGAN in order to achieve a positive impact on the segmentation performance. Specifically, we trained in turn the SPADE cGAN network by using datasets with different cardinality, respectively with 15, 20, 25, 30, 50, 75, 125, 250, 500, 750, and 1,000 images extracted from the Bonn sugar beet dataset. Then we used each trained model to generate sugar beet crop images. An example of output image from each model can be seen in Fig. 7.

Table 1: Error over six evaluation metrics between SPADE models generated samples and real samples.

Model	EMD	FID	Inception	KNN	MMD	MODE	Mean error
SPADE-15	21.74	0.16	0.95	0.49	0.56	1.38	4.21
SPADE-20	21.35	0.29	0.70	0.49	0.55	1.45	4.13
SPADE-25	17.05	0.33	0.65	0.43	0.40	1.18	3.34
SPADE-30	9.4	0.1	1.1	0.4	0.30	1.11	2.0
SPADE-50	10.1	0.3	1.2	0.28	0.235	**1.03**	2.2
SPADE-75	7.53	0.02	1.19	0.42	0.27	1.28	1.8
SPADE-125	8.73	0.23	**0.68**	0.43	0.25	1.28	1.93
SPADE-250	5.57	0.2	1.3	0.29	0.19	1.33	1.5
SPADE-500	4.1	0.16	0.92	0.2	0.14	1.082	1.1
SPADE-750	1.04	**0.02**	1.31	0.27	0.193	1.2	0,7
SPADE-1000	**0.03**	0.07	1.17	**0.09**	**0.07**	1.15	**0.43**

Table 2: Segmentation performance for the Bonnet architecture, trained on two different inputs, namely RGB and RGB + NIR, by using different training sets augmented with a varying amount of synthetic data.

	RGB				RGB+NIR			
Model		IoU				IoU		
	mIoU	Soil	Crop	Weed	mIoU	Soil	Crop	Weed
Mix-15	0.51	0.99	0.16	0.38	0.52	0.99	0.22	0.36
Mix-50	0.53	0.99	0.21	0.38	0.59	0.99	0.22	0.57
Mix-75	0.62	0.99	0.33	0.53	0.63	0.99	0.43	0.25
Mix-125	0.58	0.99	0.21	0.54	0.72	0.99	0.37	0.80
Mix-500	0.72	0.99	0.35	0.81	0.77	0.99	0.50	0.83
Mix-750	0.73	0.99	**0.41**	0.81	0.81	0.99	0.53	0.90
Mix-1000	**0.76**	0.99	0.38	**0.92**	**0.82**	0.99	**0.55**	**0.92**

We performed two kinds of evaluation for the trained models. In the first one, we used the metrics described in Section 4.2. We computed the mean of the metrics over 20 random selected sets from the real dataset. We saved these values as a reference for later comparison. Then, we computed the metrics for each model. Finally, to retrieve a single value representing the best model, we computed the mean error between the trained models metrics and the reference metrics. Table 1 shows the evaluation results. In this table, each model is named with *SPADE-N*, where *N* is replaced with the cardinality of the dataset used to train the model. Generally, the higher the number of images, the smaller the error.

In the second evaluation, we used the trained SPADE models to generate different datasets for semantic segmentation with synthetic crop, by using the proposed approach. We then augmented the real dataset with synthetic generated ones and we used such augmented datasets to train Bonnet [20], which is a state-of-the-art semantic segmentation network for precision farming. Finally, we evaluated each trained model by using part of the real dataset not used in train-

ing as test data. The results are shown in Table 2, where the Intersection over Union (IoU) and Mean Intersection over Union ($mIoU$) metrics were used [5]. Each model is named *Mix-N*, where N is replaced with the cardinality of the dataset used to train the cGAN to generate the synthetic images used in the dataset augmentation. The results show that the IoU increases by increasing the cardinality of the dataset used to train the SPADE network.

5.3 Semantic Segmentation Results

We have carried out four experiments to show that, by augmenting the training datasets with synthetic photo-realistic images, it is possible to increase the performance in crop/weed segmentation tasks.

1. In the first experiment, we augmented the Bonn dataset with synthetic photo-realistic images of sugar beet and weed plants, obtaining better results;
2. In the second experiment, we show the importance of using multi-spectral images;
3. In the third experiment, we compared the use of traditional augmentation techniques with our method, demonstrating that the use of cGAN is a winning strategy;
4. In the fourth experiment, we augmented the sunflower dataset with synthetic photo-realistic images of sunflower plants, obtaining better results.

In the following experiments, we divided the data into different, non-overlapping subsets composed by a set of images for training and a set of images for test.

Experiment 1: Augmenting the Sugar Beet Dataset. In this experiment, we trained two different types of SPADE networks. The first one was trained to generate sugar beet crops, the second one was trained to generate weeds (general types of weed, similar to those one appearing in the Bonn dataset).

Using data from both the Bonn and Stuttgart sugar beet datasets, we created four different datasets:

1. *Original*, which is a reduced version of the Bonn dataset. We took a total of 1,600 images from the Bonn dataset, randomly chosen among different days of acquisition in order to contain different growth-stages of the target crop. Then, we split it into a training set (1,000 images), a validation set (300 images), and a test set (300 images).
2. *Synthetic Crop*, composed of 1,000 images with synthetic crop generated by using our architecture.
3. *Synthetic Weed*, composed of 1,000 images with synthetic weeds generated by using our architecture.
4. *Mixed*, containing 1,000 images and composed by the union of 500 images from the Original dataset and 500 images with synthetic crop and weeds.

Table 3: Pixel-wise segmentation performance with RGB input, networks trained on four different datasets, tested on part of Bonn dataset.

SSN	Dataset	mIoU	IOU			Recall		Precision	
			Soil	Crop	Weed	Crop	Weed	Crop	Weed
Bonnet	*Synthetic Weed*	0.65	0.99	0.64	0.31	0.71	0.53	0.64	0.38
	Synthetic Crop	0.67	0.99	0.73	0.30	0.93	0.49	0.73	0.45
	Original	0.70	0.99	0.75	0.35	0.84	0.54	0.82	0.49
	Mixed	**0.76**	0.99	**0.92**	**0.38**	**0.96**	**0.58**	**0.95**	**0.56**
UNet-ResNet	*Synthetic Weed*	0.64	0.99	0.73	0.18	0.73	0.20	0.86	0.30
	Synthetic crop	0.69	0.99	0.79	0.29	0.81	0.33	0.96	0.18
	Original	0.67	0.99	0.81	0.23	**0.95**	0.25	0.95	**0.63**
	Mixed	**0.72**	0.99	**0.85**	**0.32**	0.87	**0.45**	**0.97**	0.48
U-Net	*Synthetic Weed*	0.63	0.99	0.71	0.19	0.74	0.19	0.93	0.35
	Synthetic Crop	0.65	0.99	0.76	0.20	0.81	**0.37**	0.98	0.37
	Original	0.68	0.99	0.82	0.22	0.84	0.26	0.95	0.64
	Mixed	**0.71**	0.99	**0.87**	**0.27**	**0.89**	0.28	**0.97**	**0.72**
SegNet	*Synthetic Weed*	0.61	0.99	0.60	0.14	0.72	0.18	0.86	0.47
	Synthetic Crop	0.64	0.99	0.74	0.17	0.80	0.21	0.91	0.48
	Original	0.66	0.99	0.78	0.20	0.81	0.37	**0.97**	0.37
	Mixed	**0.70**	0.99	**0.85**	**0.26**	**0.88**	**0.40**	0.96	**0.49**

For testing, we used 300 real images from the Stuttgart dataset and 300 real images (not used for training) from Bonn dataset. It is worth noting that we used the Stuttgart dataset to show the improvement in the generalization capability of the segmentation network after augmenting the training dataset with our approach. We point out that in the synthetic datasets we replaced with synthetically generated samples only the plants whose stem is totally framed into the image. For the plants that are mostly out of the frame, the original one is kept. We experimentally verified that it is necessary to have the stem of the plant roughly in the center the mask to obtain an effective synthetic image generation.

We used the four datasets described above to train four state-of-the-art semantic segmentation networks, namely U-Net [27], UNet-ResNet (U-Net with ResNet50 back-end), Bonnet [20], and SegNet [1]. An example of segmentation results is shown in Fig. 8.

To evaluate the semantic segmentation output, we used the following metrics: Per-class Intersection over Union, Mean Intersection over Union (denoted as $mIoU$), Per-class Recall, and Per-class Precision.

Table 3 shows the quantitative results of the semantic segmentation on real images held out from Bonn dataset. For all the architectures, the results show that the IoU increases by using the original dataset augmented with the synthetic ones compared to using only the original dataset. Additionally, we can see that the rate of correctly predicted crop and weed samples increased across all architectures when we used the mixed dataset for training. For example, in Bonnet architecture, the correctly predicted samples increased more than 10%

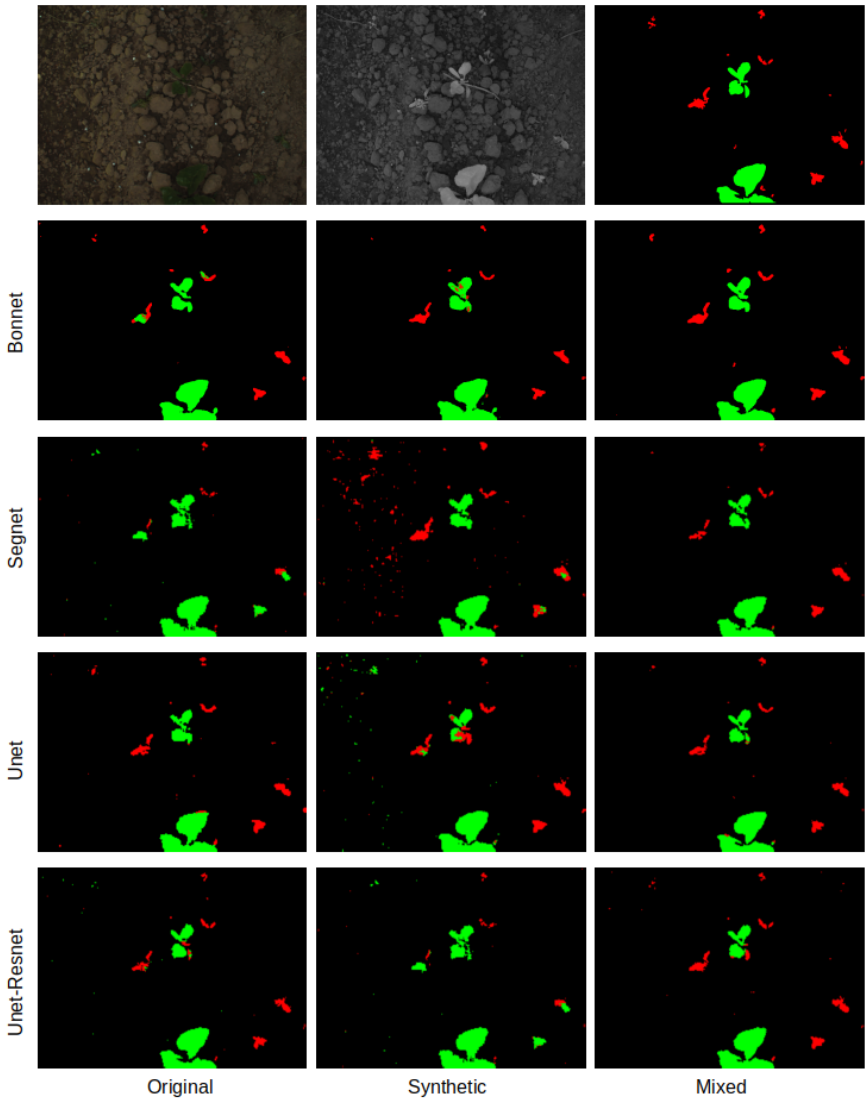


Fig. 8: Examples of a segmented image from Bonn sugar beet test set obtained by using four different segmentation networks trained with three different datasets. The first row of the image shows the input RGB, NIR images and their corresponding ground truth. The remaining rows show the segmentation results generated by the different networks on the *Original*, *Synthetic*, and *Mixed* training datasets.

in case of sugar beet, and around 4% for weed samples. Moreover, using only the synthetic dataset also leads to a competitive performance when compared to

Table 4: Pixel-wise segmentation performance, networks trained on two different inputs (RGB and RGB + NIR), tested on two different datasets.

Train set	Test set	Bonnet			UNet-ResNet			U-Net		
			IOU			IOU			IOU	
		mIoU	Crop	Weed	mIoU	Crop	Weed	mIoU	Crop	Weed
Original	*Stuttgart*	0.30	0.13	0.1	0.31	0.11	0.12	0.33	0.14	0.08
(RGB)	*Bonn*	0.70	0.75	0.35	0.67	0.81	0.23	0.68	0.82	0.22
Mixed	*Stuttgart*	0.38	0.26	0.18	0.35	0.2	0.13	0.37	0.21	0.15
(RGB)	*Bonn*	0.76	**0.92**	0.38	0.72	0.85	0.32	0.71	**0.87**	0.27
Original	*Stuttgart*	0.49	0.32	0.12	0.47	0.30	0.15	0.45	0.34	0.13
(RGB+NIR)	*Bonn*	0.77	0.85	0.45	0.31	0.85	0.35	0.69	0.84	0.24
Mixed	*Stuttgart*	**0.57**	**0.46**	**0.28**	**0.54**	**0.52**	**0.16**	**0.53**	**0.50**	**0.19**
(RGB+NIR)	*Bonn*	**0.80**	0.88	**0.55**	**0.77**	**0.92**	**0.40**	**0.74**	0.85	**0.37**

using only the original one. In the case of the UNet-ResNet architecture, using only the synthetic dataset overcomes the performance obtained by using only the original one. We can notice also that crop has a more positive impact on the dataset used in semantic segmentation and that is because the quality of synthetically generated crops are better than the quality of synthetic generated weeds. This behavior stems from the fact that weeds have more diverse types and shapes, which makes the task of cGAN of capturing weed style distribution more challenging.

Experiment 2: The Importance of Multi-Spectral Images. To show the contribution of having both multi-spectral and synthetic data augmentation, we considered four different training sets, i.e., *Original* and *Mixed* containing RGB images only and *Original* and *Mixed* containing both RGB and NIR images. We used the Stuttgart and Bonn datasets as test data. Table 4 shows the segmentation results for this experiment. For all the tested architectures, the segmentation capability improves when using the *Mixed* dataset, i.e., when the dataset containing real images is augmented with synthetic data. This supports the idea of creating artificial samples to improve the segmentation performance.

Moreover, the results in Table 4 show that using the *Mixed* RGB plus NIR dataset during the training process leads to a better performance. In fact, the segmentation performance increases in all the considered setups. This proves our claim that also the NIR channel generated using our approach improves the segmentation capability of all the convolutional network architectures used in our experiments.

Experiment 3: Comparison with Traditional Augmentation Techniques. Basic image manipulations to obtain training data augmentation include rotation, shifting, flipping, zooming, and cropping. Also texture manipulations like

Table 5: Segmentation results of Bonnet architecture, trained on four different datasets, tested on Bonn test dataset

Augmentation Strategy	IoU		
	mIoU	Crop	Weed
Basic augmentation	0.71	0.76	0.37
Texture augmentation	0.73	0.79	0.40
Ours + Basic augmentation	**0.78**	**0.93**	**0.43**
Ours + Texture manipulation	0.66	0.80	0.19

Gaussian and median blurring, noise injection, and contrast and brightness variation can be used to augment the available data. The aim of this experiment is to show the effectiveness of our method over the traditional augmentation strategies. We prepared four training datasets as follows.

- **Basic augmentation**: 1,000 original images augmented with 1,000 images using basic image operations.
- **Texture augmentation**: 1,000 original images augmented with 1,000 images using texture manipulations.
- **Ours + Basic augmentation**: 1,000 original images augmented with 500 images using our strategy plus 500 images as in basic augmentation;
- **Ours + Texture augmentation**: 1,000 original images augmented with 500 images using our method plus 500 images processed with texture manipulation strategies.

Results are reported in Table 5. We can see that the best results are achieved by the model trained on the dataset augmented using both our method and a basic augmentation, while augmenting the dataset with our method plus texture manipulations cause a drop in the mIoU.

Experiment 4: Augmenting the Sunflower Dataset. In this experiment, we used 500 images to train SPADE networks to generate sunflower crop, the second one trained to generate weeds (general types of weeds similar to those in Bonn dataset). In this experiment we used sunflower images. We created three different datasets:

1. *Original*, We took a total of 500 images from the Sunflower dataset, randomly chosen among different days of acquisition in order to contain different growth-stages of the target crop. Then, we split it into a training set (350 images), and a test set (150 images).
2. *Synthetic Crop*, composed of 350 images with synthetic crop generated by using our architecture.
3. *Mixed*, containing 350 images and composed by the union of 175 images from the Original dataset and 175 images with synthetic crops.

We used the three datasets described above to train three state-of-the-art semantic segmentation networks, i.e., Bonnet, U-Net, and UNet-ResNet. An example of segmentation results is shown in Fig. 9. Table 6 shows the quantitative

Table 6: Pixel-wise segmentation performance for Sunflower Dataset, networks trained on two different inputs (RGB and RGB + NIR).

Dataset	SSN	mIoU	IOU			Recall		Precision	
			Soil	Crop	Weed	Crop	Weed	Crop	Weed
Synthetic	*Bonnet*	0.76	0.99	0.84	0.46	0.94	0.61	0.88	0.67
Crop	*U-Net*	0.51	0.98	0.038	0.51	0.039	0.51	0.57	0.98
(RGB)	*Unet-resnet*	0.53	0.98	0.05	0.59	0.05	0.63	0.32	0.89
Synthetic	*Bonnet*	0.83	0.99	0.84	0.66	0.97	0.81	0.85	0.78
Crop	*U-Net*	0.701	0.99	0.69	0.41	0.81	0.45	0.83	0.87
(RGB+NIR)	*UNet-ResNet*	0.704	0.99	0.65	0.47	0.73	0.52	0.86	0.82
Original	*Bonnet*	0.70	0.99	0.82	0.30	0.94	0.61	0.86	0.23
	U-Net	0.39	0.97	0.15	0.031	0.17	0.04	0.72	0.10
(RGB)	*UNet-ResNet*	0.43	0.98	0.28	0.04	0.34	0.06	0.65	0.08
Original	*Bonnet*	0.80	0.99	0.78	0.62	0.87	0.82	0.88	0.71
	U-Net	0.64	0.99	0.54	0.38	0.70	0.42	0.83	0.40
(RGB+NIR)	*UNet-ResNet*	0.66	0.99	0.60	0.43	0.75	0.49	0.88	0.90
Mixed	*Bonnet*	0.78	0.99	0.87	0.48	0.95	0.59	0.91	0.73
	Unet	0.59	0.98	0.64	0.12	0.71	0.12	0.88	0.98
(RGB)	*Unet-resnet*	0.60	0.98	0.67	0.15	0.74	0.14	0.90	**0.97**
Mixed	*Bonnet*	**0.86**	**0.99**	**0.88**	**0.69**	**0.97**	**0.85**	**0.90**	0.79
	U-Net	0.69	0.99	0.68	0.40	0.80	0.43	0.83	0.87
(RGB+NIR)	*UNet-ResNet*	0.72	0.99	0.70	0.48	0.77	0.51	0.88	0.87

results of the semantic segmentation on real images held out from sunflower dataset. In all cases, for testing we used 150 real images from Sunflower dataset not used during training. Also in the case of this dataset for all architectures the results show that the mIoU increases by using the original dataset augmented with the synthetic ones, as compared to using only the original dataset. Moreover, for this dataset architectures trained on synthetic data perform better than when trained on real data only.

All the datasets generated using the approach described in this work are publicly available and can be downloaded from:

`https://bit.ly/3hHenpE`

5.4 Statistical Analysis

To further support the significance of our results, a statistical analysis have been carried out. The null hypothesis, which we want to reject, says that there is no difference in the segmentation results if we use a dataset augmented with synthetic data or a dataset without. Since the null hypothesis is presumed to be true until the data shows enough evidence that it is not, we show here that a model built from a dataset augmented with synthetic data generates better results in the vast majority of the cases with respect to a model trained without them.

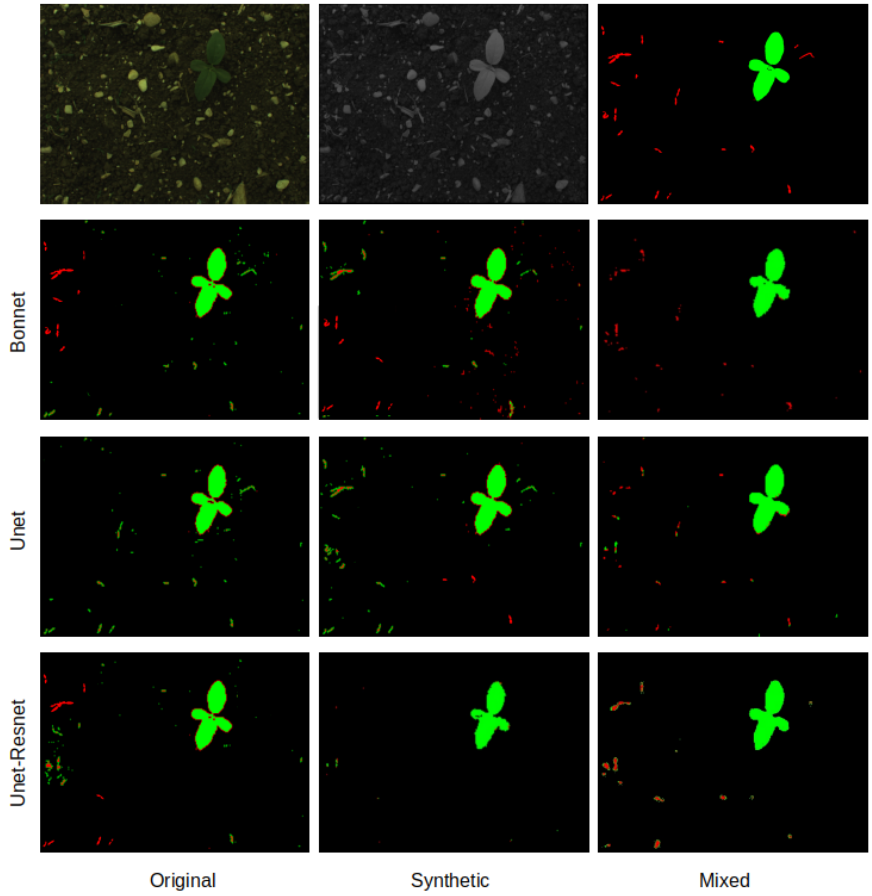


Fig. 9: Examples of a segmented image from sunflower test set obtained by using three different segmentation networks trained with three different datasets. The first row of the image shows the input RGB, NIR images and its corresponding ground truth. The remaining rows show the segmentation results generated by the different networks on the *Original*, *Synthetic*, and *Mixed* training datasets.

The analysis takes into account the Bonnet network because it was globally the best in our tests. We trained Bonnet on two different RGB sugar beet datasets, namely the *Original* and *Mixed* ones. We tested the two models, named O and M in Table 7, on 300 test images from the Bonn dataset, comparing the results on three evaluation metrics: *Accuracy*, Dice similarity coefficient ($DICE$), and intersection over union (IoU). Although *Accuracy* is easy to calculate and understand, it is not useful when the foreground and background classes are extremely imbalanced, i.e., when a class dominates the image and the other covers only a small portion of the image, which is the case in our scenario. Better

Table 7: Statistical analysis for the Bonnet architecture trained on two different sugar beet datasets and tested on 300 images.

Evaluation metrics	# of images where M is better than O	# of images where O is better than M
Accuracy	284	16
DICE	289	11
IoU	290	10

metrics for dealing with the class imbalance issue are DICE and IoU. The Dice score reflects both size and localization agreement, more in line with perceptual quality compared to pixel-wise accuracy [2].

Results are shown in Table 7. Our method (model M) provides better results in 94.6% of images using Accuracy, 96.3% using DICE, and 96.6% using IoU, with an average value of 95.3%. Since in about 95% of the cases we have better results with synthetic augmentation, we can reject the null hypothesis.

5.5 Comparison with a Non-Conditional GAN

The proposed system leverages a cGAN to generate data to be used to augment the available training data, using plant masks as an essential prerequisite. But what if we use a non-conditional generic Generative Adversarial Networks (GAN) [9] modified to generate both the images and the segmentation masks? To answer this question, we designed a specific experiment in which a generic GAN has been modified and trained so to generate *both* RGB images of crop and soil *and* the related segmentation masks (i.e., the plant and soil masks).
We exploited the GAN architecture presented in [26], modifying it in order to generate 4-channels images (i.e., the RGB image plus the segmentation mask). The segmentation masks of the training dataset defines one class (i.e., crop) with the value 1 and the other class (i.e., soil) with the value -1; we converted the (continuous) masks generated by the GAN simply by applying a threshold operator with threshold 0. As conventional GANs, the generator network takes noise as input. Early experiments showed poor results in generating entire agricultural scenes (crop, weeds and soil) with accurate segmentation masks at full image resolution. In particular, the generated masks were qualitatively inadequate and inconsistent with real plants. We believe that this fact mostly derives from the limited size of the available training datasets (around 1,000 images in ours case). Hence, we decided to simplify the task, focusing on 256×256 patches depicting single instance of crops, and soil as background. With this setup. the GAN results were apparently convincing (e.g., see Fig. 10). The dataset used to train the GAN was composed of 2,000 patches with related segmentation masks taken from the Bonn sugar beet dataset.

We created three datasets:

(a) RGB. (b) Mask.

Fig. 10: Example of a sugar beet generated by a non-conditional GAN: (a) RGB image; (b) Segmentation mask obtained by thresholding the mask generated by the GAN.

Table 8: Pixel-wise segmentation performance for Bonnet architecture, trained on three different datasets.

Model	IoU		
	mIoU	Soil	Crop
Original	0.85	0.94	0.76
Original+Ours	**0.94**	**0.98**	**0.89**
Original+GAN	0.61	0.88	0.34

- *Original*: 2,000 crop and soil patches extracted from the Bonn dataset
- *Original + Ours*: *Original* dataset augmented with 500 patches generated by our approach
- *Original + Ours*: *Original* dataset augmented with 500 patches generated by the non-conditianl GAN as previously described

We trained the Bonnet network on such datasets, testing the segmentation results on 300 test images from the Bonn dataset: the results are reported in Table 8. Consistent with the previous results, our method allows to improve the performance compared to the *Original* dataset (9% increase in mIoU). Conversely, the model trained with the images generated by the GAN achieves poor performance. This result is mainly ascribable to the shape (i.e., the mask) and texture of the generated plants from the GAN, which do not comply to those of real plants.

6 Conclusions

This paper introduces a data augmentation strategy that leverages a cGAN to generate entire agricultural scenes by synthesizing only the most relevant objects for segmentation purposes. The core of the proposed approach lies in exploiting

the shapes of real objects to condition the trained generative models. The existing shapes are extracted from real-world labeled images. In addition, the generation process also synthesizes the NIR channel. The synthetically augmented dataset, obtained in this way, can then be used to train a semantic segmentation network. We applied this method to the crop/weed segmentation problem. As a further contribution, we also introduce and made publicly available with this paper a new crop/weed segmentation dataset, the Sunflower Dataset. Two kinds of quantitative evaluation have been carried out. In the first one, we test the cGAN generalization properties. Our experiments prove that with a small number of images we are able to generate good synthetic plant samples. The second evaluation aims to demonstrate that the cGAN augmented datasets can improve the performance of different state-of-the-art segmentation architectures. The results show that the segmentation quality increases by using the original dataset augmented with the synthetic ones, with respect to using only the original dataset. We believe that our method can serve as a valid tool for creating training frameworks for segmentation problems, allowing to improve segmentation performance, while reducing the amount of required labeled data.

References

1. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. CoRR abs/1511.00561 (2015)
2. Bertels, J., Eelbode, T., Berman, M., Vandermeulen, D., Maes, F., Bisschops, R., Blaschko, M.B.: Optimizing the dice score and jaccard index for medical image segmentation: Theory and practice. Medical Image Computing and Computer Assisted Intervention – MICCAI 2019 p. 92–100 (2019)
3. 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 (2017)
4. 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 (2017)
5. Everingham, M., Eslami, S.M., Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. Int. J. Comput. Vision 111(1), 98–136 (Jan 2015)
6. Fawakherji, M., Youssef, A., Bloisi, D., Pretto, A., Nardi, D.: Crop and weeds classification for precision agriculture using context-independent pixel-wise segmentation. In: 3rd IEEE International Conference on Robotic Computing, IRC 2019, Naples, Italy, February 25-27, 2019. pp. 146–152 (2019)
7. Giuffrida, M.V., Scharr, H., Tsaftaris, S.A.: ARIGAN: Synthetic arabidopsis plants using generative adversarial network. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 2064–2071 (2017)
8. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)

9. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 2672–2680. Curran Associates, Inc. (2014)
10. Han, C., Hayashi, H., Rundo, L., Araki, R., Shimoda, W., Muramatsu, S., Furukawa, Y., Mauri, G., Nakayama, H.: Gan-based synthetic brain mr image generation. In: 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018). pp. 734–738 (2018)
11. Haug, S., Michaels, A., Biber, P., Ostermann, J.: Plant classification system for crop/weed discrimination without segmentation. In: IEEE winter conference on applications of computer vision. pp. 1142–1149. IEEE (2014)
12. Lee, W., Alchanatis, V., Yang, C., Hirafuji, M., Moshou, D., Li, C.: Sensing technologies for precision specialty crop production. Computers and Electronics in Agriculture 74(1), 2 – 33 (2010)
13. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3431–3440 (2015)
14. 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 (2018)
15. Lottes, P., Khanna, R., Pfeifer, J., Siegwart, R., Stachniss, C.: Uav-based crop and weed classification for smart farming. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). pp. 3024–3031 (2017)
16. Lottes, P., Hörferlin, M., Sander, S., Stachniss, C.: Effective vision-based classification for separating sugar beets and weeds for precision farming. Journal of Field Robotics 34(6), 1160–1178 (2017)
17. Madsen, S.L., Dyrmann, M., Jørgensen, R.N., Karstoft, H.: Generating artificial images of plant seedlings using generative adversarial networks. Biosystems Engineering 187, 147 – 159 (2019)
18. McCool, C., Perez, T., Upcroft, B.: Mixtures of lightweight deep convolutional neural networks: Applied to agricultural robotics. IEEE Robotics and Automation Letters 2(3), 1344–1351 (2017)
19. Milioto, A., Lottes, P., Stachniss, C.: Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 2229–2235 (2018)
20. 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)
21. Mirza, M., Osindero, S.: Conditional generative adversarial nets. CoRR abs/1411.1784 (2014)
22. Mortensen, A.K., Dyrmann, M., Karstoft, H., Jørgensen, R.N., Gislum, R., et al.: Semantic segmentation of mixed crops using deep convolutional neural network. In: Proc. of the International Conf. of Agricultural Engineering (CIGR) (2016)
23. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2337–2346 (2019)
24. Potena, C., Nardi, D., Pretto, A.: Fast and accurate crop and weed identification with summarized train sets for precision agriculture. In: International Conference on Intelligent Autonomous Systems. pp. 105–121. Springer (2016)

25. Pretto, A., Aravecchia, S., Burgard, W., Chebrolu, N., Dornhege, C., Falck, T., Fleckenstein, F., Fontenla, A., Imperoli, M., Khanna, R., Liebisch, F., Lottes, P., Milioto, A., Nardi, D., Nardi, S., Pfeifer, J., Popović, M., Potena, C., Pradalier, C., Rothacker-Feder, E., Sa, I., Schaefer, A., Siegwart, R., Stachniss, C., Walter, A., Winterhalter, W., Wu, X., Nieto, J.: Building an aerial-ground robotics system for precision farming: An adaptable solution. IEEE Robotics & Automation Magazine (2020)
26. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks (2016)
27. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. CoRR abs/1505.04597 (2015)
28. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) 115(3), 211–252 (2015)
29. Sa, I., Chen, Z., Popović, 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 (2018)
30. Sixt, L., Wild, B., Landgraf, T.: Rendergan: Generating realistic labeled data. Frontiers in Robotics and AI 5, 66 (2018)
31. Sixt, L., Wild, B., Landgraf, T.: Rendergan: Generating realistic labeled data. Frontiers in Robotics and AI 5, 66 (2018)
32. Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1–9 (2015)
33. Ustin, S.L., Jacquemoud, S.: How the Optical Properties of Leaves Modify the Absorption and Scattering of Energy and Enhance Leaf Functionality, pp. 349–384. Springer International Publishing (2020)
34. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8798–8807 (2018)
35. Xie, J., Kiefel, M., Sun, M., Geiger, A.: Semantic instance annotation of street scenes by 3D to 2D label transfer. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3688–3697 (2016)
36. Xu, Q., Huang, G., Yuan, Y., Guo, C., Sun, Y., Wu, F., Weinberger, K.Q.: An empirical study on evaluation metrics of generative adversarial networks. CoRR abs/1806.07755 (2018)