Sim2real flower detection towards automated Calendula harvesting

Wout Vierbergen[a,b,*] Axel Willekens[a,c], Donald Dekeyser[a], Simon Cool[a], Francis wyffels[c]

[a]Technology and Food Science Unit, Flanders Research Institute for Agriculture, Fisheries and Food (ILVO), Burg. van Gansberghelaan 115, Merelbeke, 9820, Belgium

[b]MeBioS, Department of Biosystems, KU Leuven, Kasteelpark Arenberg 30, Leuven, 3001, Belgium

[c]AI and Robotics Lab (AIRO), IDLab, Ghent University - imec, Technologiepark-Zwijnaarde 126, Zwijnaarde, 9052, Belgium

---

Abstract

Deep learning has gained a lot of attention in the last decade for its use in computer vision. However, a barrier to use deep learning in an agricultural context is the need for large datasets. Agricultural processes are situated in uncontrolled environments, making data collection even harder than in other contexts. Factors such as plant growth, weather conditions, and illumination

* Corresponding author

Email addresses: wout.vierbergen@ilvo.vlaanderen.be (Wout Vierbergen), axel.willekens@ilvo.vlaanderen.be (Axel Willekens), donald.dekeyser@ilvo.vlaanderen.be (Donald Dekeyser), simon.cool@ilvo.vlaanderen.be (Simon Cool), francis.wyffels@ugent.be (Francis wyffels)

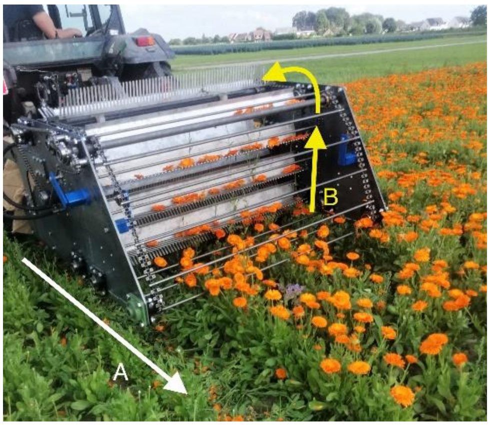
are largely uncontrolled, making it hard to collect all possible variations in a dataset. This study demonstrates how synthetic generated data can aid to overcome the current barrier and it exemplifies this in the context of automating the detection and localisation of Calendula flowers. To this end, a pipeline was created that utilises photogrammetry and a flower field simulator to create synthetic data of a flower field. Next, the synthetic data is used to train a deep neural network to detect flowers and the transfer from simulation to reality (sim-to-real) is demonstrated on real data. Although the flower detector has not been trained on real data, it reaches an F1 score of up to 86% on the test sets of real data. Subsequently, a stereo vision camera system utilises this detection model to accurately determine the 3D positions of the flowers. The localisation results in an error of 6,9 ± 5,1 mm for the prediction of the flower height. In conclusion, leveraging the potential of synthetic data and sim-to-real capabilities can lower the costs of collecting large datasets in uncontrolled environments and can accelerate the development of precision agricultural applications.

Keywords: Synthetic data, Sim-to-real, Deep learning, Precision agriculture

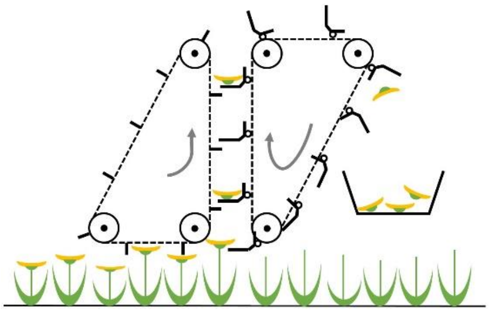
1. Introduction

The flower of a Calendula plant (Calendula officinalis L.) has many interesting and valuable properties. The Calendula flower can either be consumed fresh or used as a colourant in foods. Oil from the flowers, in turn, can be used in medical ointments and cosmetics. In addition, the seed oil is a coveted substance for paints and coatings (Khalid, 2012). Due to its properties and the wide habitat of the flower, the Calendula flower is of interest to farmers in large regions of the world. Currently, however, Calendula flowers are mainly harvested manually. This results in high labour costs which makes the cultivation of Calendulas economically not feasible in many countries.

In the need for mechanical harvesting methods for the Calendula flower, different mechanical harvesting methods have been proposed over the past decades. Willoughby et al. (2000) proposed two different systems, both based on rotating pairs of picking fingers. Other work was performed by Veselinov et al. (2014), who harvested Calendula flowers with a virtual rotating combtype harvester. More recently, a similar mechanism is used in the work of Wang et al. (2021) on the design, simulation and test of Chrysanthemum (Dendranthema morifolium Ramat.) picking machine. Lastly, Fig. 1 shows another mechanical prototype for a Calendula harvester that has been developed by Flanders Research Institute for Agriculture, Fisheries and Food (ILVO). In contradiction to the other designs, in this design, the combs do not rotate but move in a vertical way to pick flowers. This is a similar movement to manual flower picking. In all these works, the height of the harvester is fixed at one position and is not adjusted to the actual height of the harvested flowers. Since differences in flower height occur at various positions in a field, the studies notice a decrease in harvest efficiency in case the height of the harvester is not properly adjusted to the height of the flowers at a certain position in the field (Veselinov et al., 2014; Wang et al., 2021; Willoughby et al., 2000).



a) While the machine moves in the direction of A, the flowers are picked by combs moving as indicated by arrow B.



b) Side view of harvesting machine. Height adjustment of the machine relative to the flower heads determines the harvest efficiency and quality.

Figure 1: Harvesting machine for Calendula flowers developed by ILVO.

One way to improve the harvest efficiency is by equipping these mechanical harvesters with robotic components to perceive the flowers, detect their height, and automatically adjust the height of the harvester (Bechar and Vigneault, 2016). To detect the flowers, machine vision can be utilised (Mavridou et al., 2019). In recent years, several studies have explored the use of machine vision techniques to detect flowers (Dias et al., 2018; Wang et al., 2022), fruits (Rahnemoonfar and Sheppard, 2017; Sa et al., 2016) or weeds (Hasan et al., 2021; Picon et al., 2022).

The use of deep learning for computer vision and object detection has gained a lot of attention in the last decade. Supervised deep learning technologies outperform older computer vision techniques (Kamilaris and Prenafeta-Boldú, 2018; Zhang et al., 2020). With these developments, object and keypoint detectors based on convolutional neural networks (CNN) such as YOLO (Redmon et al., 2016) and CenterNet (Zhou et al., 2019) have become state of the art and are capable of detecting learned objects in real-time in challenging conditions.

However, training a supervised deep learning algorithm often requires the availability of large, labeled datasets for training. This is especially true in the case of agricultural applications, where it is challenging to handle all possible variations that can occur, for example in illumination, background, arrangement of the objects, occurrence of weeds, and growth stage of the plants. Moreover, this makes data collection costly and creates a bottleneck for the application of deep learning in an agricultural context (Kamilaris and Prenafeta-Boldú, 2018; Roh et al., 2021). Public available datasets such as ImageNet (Deng et al., 2009) and MS COCO (Lin et al., 2015) have been a huge contribution to the computer vision community in the development of object detection/segmentation models. Yet, these datasets are very generic and do not translate well to agricultural applications. The lack of public datasets targeted to specific agricultural applications does not alleviate this bottleneck for most precision agricultural applications (Lu and Young, 2020).

To eliminate this bottleneck, the use of synthetically created data has gained a lot of attention in the past few years (de Melo et al., 2022; Nikolenko, 2019; Tobin et al., 2017; Tremblay et al., 2018). Synthetic data generation has the advantages that it is possible to quickly generate scenes that are hard to capture in reality, is inherently accompanied by pixel-perfect labels, and makes quick iterations possible. In the case of agricultural applications, synthetic data generation makes it possible to create data with high variability. Environmental variables such as illumination, plant growth, shape, and texture can be determined arbitrarily in this process. Due to the large possible variability in an agricultural process and strong seasonal dependencies, Rizzardo et al. (2020) argue that the use of virtual environments to simulate these conditions and agricultural processes is a necessity for the development of agricultural robots.

However, a challenge in using synthetic data is the transfer to the real world (sim-to-real). Generally, this is overcome by applying (structured) domain randomisation to the scene (Prakash et al., 2019; Tobin et al., 2017).

Synthetic image data can be generated in various ways. Rahnemoonfar and Sheppard (2017) created synthetic training data to count tomatoes by simply generating a green/brown background and drawing random red dots on top of the background (Rahnemoonfar and Sheppard, 2017). In other work the *Cut, Paste, and Learn* (Dwibedi et al., 2017) approach is exploited to generate new images by combining parts of different RGB images (Picon et al., 2022; Wang et al., 2022). This method, however, is limited in the kinds of variation that can be introduced. Different 3D orientations of the individual objects and lighting effects such as shadows can not be introduced with this approach. To simulate more realistic scenes, 3D models of plants can be placed in a virtual environment such as a game engine (Qiu and Yuille, 2016). To create these 3D models of biological material, photogrammetry has been proven successful in plant reconstruction (Andújar et al., 2018). Further, L-systems offer promising results in generating realistic models of plants in different growth stages (Cieslak et al., 2022).

In this work, we generate a synthetic dataset of Calendula flowers based on a few 3D models of the plants and validate its purpose for the localisation of the flowers.

The contributions in this work are threefold:

1. We present a pipeline to generate synthetic data of agricultural processes with the use of photogrammetry and a game engine.
2. A Calendula flower detector based on a CNN is trained on synthetic data and validated on a test set of real Calendula images (sim-to-real

transfer). This flower detector, combined with stereo vision, enables the localisation of the flowers to automatically adjust the height of harvesters to increase harvest efficiency.

3. Lastly, all collected and generated data is made available on Zenodo as a contribution to future research on precision agriculture (Vierbergen et al., 2022).

2. Materials and methods

The different steps in the generation and use of synthetic data for flower detection and localisation can be divided into three categories: synthetic data generation, training of flower detection model, and sim-to-real evaluation. These subdivisions and their corresponding steps are shown in Fig. 2. To create synthetic plants, first and foremost, 3D models of the Calendula are created using photogrammetry. After decomposing the flowers and leaves into different 3D models, these are used as assets in a flower field simulator together with images of soil with weeds. By using these assets and a simulation framework in the flower field simulator, synthetic data can be generated. This synthetic dataset is subsequently used to train and immediately evaluate a deep neural network. Without any kind of transfer learning, the trained network is finally evaluated on images captured in an uncontrolled, outdoor environment.

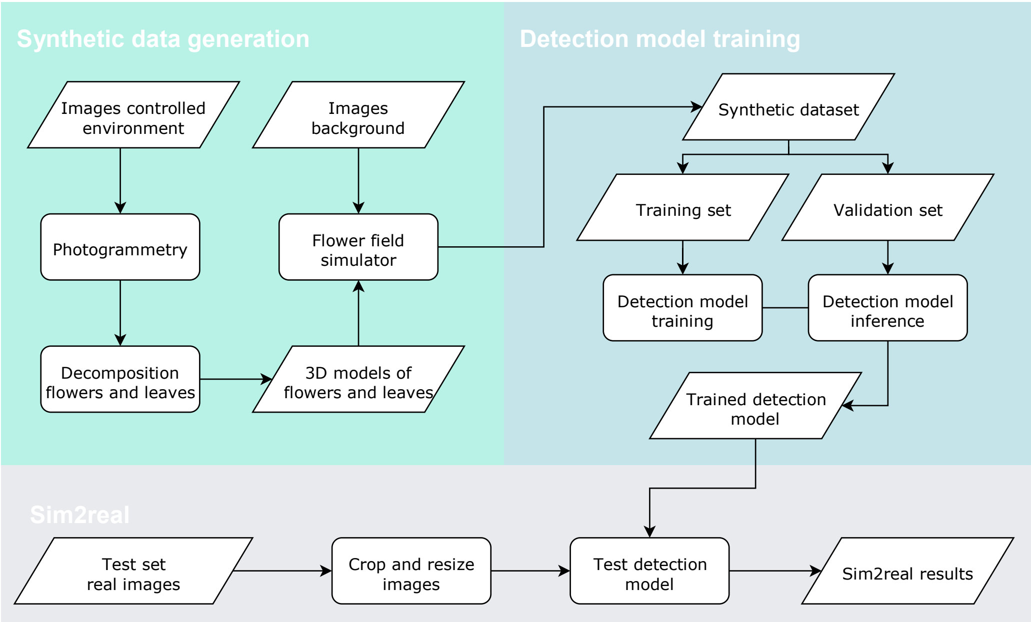


Figure 2: Visualisation of different steps sim-to-real pipeline: synthetic data generation, detection model training and sim-to-real validation.

Before discussing the synthetic data generation, section 2.1 expands on the data that has been collected for this study. Next, section 2.2 describes our pipeline to generate synthetic agricultural data. The section is followed by a description of the flower detection system in section 2.3 and the localisation of the flowers in section 2.4.

2.1. Data collection

A total of three different datasets were compiled for this study. Two datasets consist of images of real Calendula flowers captured in respectively an uncontrolled and a controlled environment. A third dataset is synthetically generated and will be discussed in the next section.

2.1.1. *Field data*

The first dataset consists of images of Calendula flowers on the field when they would be harvested. The images in this dataset are collected in an outdoor and strongly varying environment at different moments in time under different weather conditions.

To compile this dataset, an Intel RealSense D415 (Intel Corporation, Santa Clara, USA) depth camera was used to collect both RGB and depth images of Calendula plants. By mounting the camera on a trolley or tractor, a fixed camera height and steady horizontal speed of about 3 km h^{-1} were obtained. Figure 3 illustrates this setup. Images were taken at an interval of one second. By capturing images of five different fields spread over seven different moments in time, the dataset includes a wide range of variety regarding the moment of capture, location, weather conditions, and lighting. At the different locations, Orange Beauty was the most prominent cultivar, although more than 15 different cultivars are included in the dataset. The tables in Appendix A give a detailed overview of the location, moment of capture, and cultivars in the dataset.

a) b)

Figure 3: Intel RealSense D415 sensor mounted on tractor (a) and trolley (b) to capture field data.

By varying the height and the pitch angle of the camera, additional variation was introduced. The RGB images were stored with a resolution of 1920*1080 in JPEG format and the aligned depth images with a resolution of 1280*720 pixels. The flowers in the images were annotated with bounding boxes using makesense.ai[1] software. Additionally, the heights and diameters of the flowers were measured in six different fields.

2.1.2. Photo booth

To create 3D models of the Calendula with photogrammetry, five plants were placed on a rotating platform in front of a white background. These plants were photographed from 100 to 150 points of view with a Canon 600D DSLR camera using a Canon EF-S 18-135 mm lens (Canon

[1] https://makesense.ai

Inc., Tokyo, Japan). The photographed plants were bought at a local florist and of an unknown cultivar. Figure 4 shows the used setup and some of the resulting images.

2.2. *Synthetic data pipeline*

Our pipeline to generate synthetic data of Calendulas consists of two steps. First, photogrammetry was used to create 3D models of a Calendula. Subsequently, a flower field simulator makes use of these assets to create a virtual Calendula field of which RGB images are captured. These images represent, synthetically, a Calendula field.

2.2.1. *Photogrammetry*

With photogrammetry, it is possible to extract 3D information from RGB images and reconstruct a virtual representation of the object. By utilising the images of a Calendula as captured in section 2.1.2, 3D models of Calendulas can be created. To this end, Agisoft Metashape (v1.5.5.9097, Agisoft LLC, St. Petersburg, Russia) was used.

After aligning the images in Agisoft Metashape, a mesh of the Calendula plant was generated using depth maps as a source, and with parameters for quality and face count set to 'high'.

A 3D mesh produced by Metashape consists of about 12 million surfaces, resulting in an Object file of about 1.55 GB for each model. To reduce the file size, the meshes were decimated to 50.000 surfaces. This reduces the file size to about 6.5 MB for an individual model, as Fig. 4 visualises.

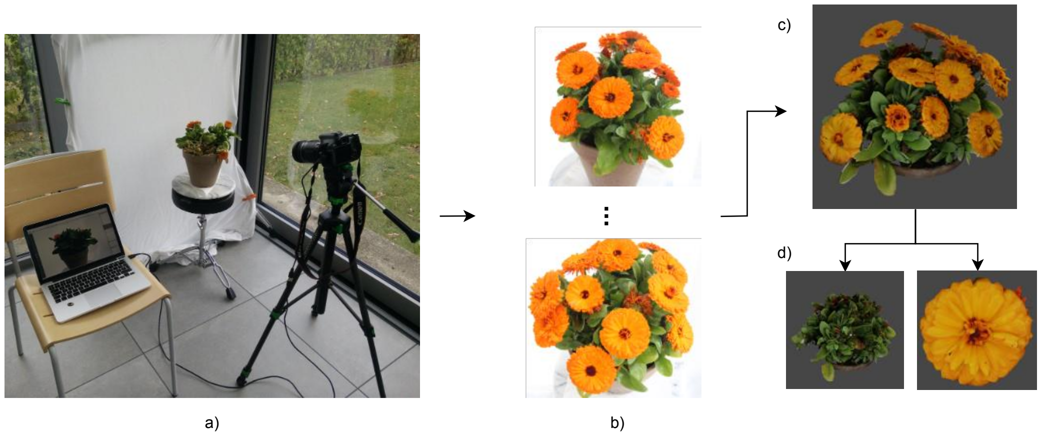


Figure 4: Creation of 3D models. Photographing a plant in front of a white background (a) results in RGB images (b) that can be used to create a 3D model of the plant (c). The model is subsequently decomposed in leaf and flower parts (d).

As a final step, the flowers and leaves of the model were decomposed and stored in different *Object* files using Blender (version 2.93.1, The Blender Foundation, Amsterdam, The Netherlands). All flowers were repositioned with their centres at the origin of the coordinate system, the leaves with their bottoms.

2.2.2. Flower field simulator

To generate the synthetic images, the Unity Perception package was used. The open-source package, developed by Unity Technologies, extends the Unity Editor and engine components to generate annotated images for computer vision tasks (Borkman et al., 2021). By using this framework, a scene was created which is made up of different layers, each filled with a certain object type. From bottom to top, the layers in the flower field simulator represent the background, leaves, flowers, illumination, and a camera. These layers are showed in Fig. 5.

Figure 5: Simulation of a flower field in Unity to generate synthetic images. The scene consists of different layers. From bottom to top: (1) background images of weeds, (2) Calendula leaves, (3) Calendula flowers, (4) point lights, and (5) a camera.

The background layer displays the soil, weeds, and shadows in the images. The layer is composed of a random selection of images from two datasets. To start with, 114 random images from the DeepWeeds dataset (Olsen et al., 2019) were selected. Since the DeepWeeds dataset is captured in Australia, 114 images were added to increase the variety and cover Belgian weeds as well.

These additional images were collected by unmanned aerial vehicle (UAV) flights above corn fields covered with weeds at Merelbeke, Belgium. These UAV flights were performed with a DJI Matrice 600 (DJI, Shenzhen, Guangdong,CHN) equipped with a Ronin MX gimbal (DJI, Shenzhen, Guangdong, China) and RGB Sony a7R III camera (42.4 MP, mirrorless) (Sony, Minato, Tokyo, Japan), with a 135 mm lens, type Carl Zeiss Batis 135 mm f2.8 (Zeiss, Oberkochen, Baden-Württemberg, Germany). To further process the images, these images were tiled into tiles of 1024 by 1024. The weeds in the images were not determined.

In the background layer, a random selection of these 228 images was placed at random positions with a random rotation and small random variation in height.

Above this background layer, the leaves of the Calendula were rendered. For each frame, a random selection of the seven leaf models was positioned at random places with a random rotation along all axis. The tilt angle was kept between -60° and +60° so that the flowers are not shown completely sideways or upside down. Each leaf model could occur zero, one, or multiple times in a single frame.

The flowers are rendered on the third layer. With the same randomisation as in the previous layer, the flower assets are positioned randomly in this layer, adding a random variation in height.

In open fields, a wide variety of light conditions occurs. To simulate this, a layer with irregularly positioned point lights was added. By varying the number of point lights positioned in this layer and their position, the scene is irregularly illuminated.

Positioned atop these layers is a Perception Camera (Borkman et al., 2021), which captures an image of the scene along with its corresponding annotations. This camera is positioned at the centre of the layer with a random variation in tilt angle and height. The resolution of the Perception Camera is set to 512*512 pixels to match the input of the detection network, as described in the next section.

2.3. Flower detection

To detect the flowers in a given image, we made use of deep learning. The architecture of the implemented model was inspired by CenterNet, a deep neural network for object detection with an excellent performance in both speed and accuracy (Zhou et al., 2019).

2.3.1. Architecture

To predict the position of the centre of the flowers, we are interested in detecting the centrepoint of a flower. To this end, we implemented a network with a ResNet-18 (He et al., 2016) backbone as used in CenterNet (Zhou et al., 2019). Since a prediction of the flower size was not needed, the output head that predicts the width and height of the bounding box around the object was not implemented. The offset head was eventually left out from the implementation since we noticed no significant improvement upon the predicted centre coordinates as obtained from the heatmap output. Figure 6 visualises the resulting network architecture.

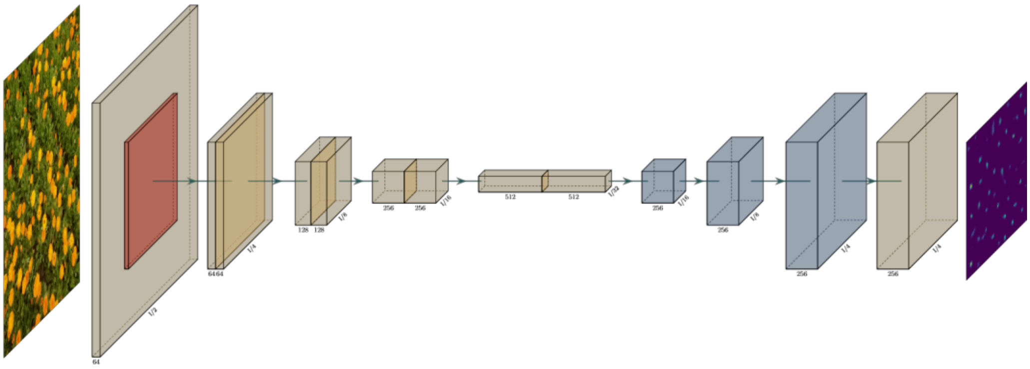


Figure 6: Visualisation of used detection network based on the CenterNet architecture.

This network takes an image $I \in [0,255]^{W \times H \times 3}$ with width W and height H as input and generates a keypoint heatmap $\hat{Y} \in [0,1]^{\frac{W}{R} \times \frac{W}{R}}$ as output, where R represents the output stride. For the experiments, the input image size was set to 512 by 512, the output resolution to 128 by 128 (output stride R = 4). It can be noticed that there is only one output class for $\hat{Y}$, namely one of the flowers.

Comparing different loss functions, a binary cross entropy (BCE) loss resulted in significantly higher results in both precision and recall compared to a focal loss (Lin et al., 2020). All results discussed in the following sections are obtained using the BCE loss L:

$$L = \frac{-1}{N} \sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i),$$

where $\hat{y}$ is the predicted heatmap and y the ground truth heatmap.

The final prediction of the centre points is obtained by applying a 3x3 maxpool to the heatmap (Zhou et al., 2019).

2.3.2. Training and validation

To make the transition from simulated to real images, the detection model was trained on 15,000 synthetically generated images (see section 2.2). Validation during training was done on a fixed set of 250 synthetic images which were not included in the training set. Finally, a trained model was tested on annotated images taken in an outdoor environment (see section 2.1.1). For training, we varied three hyperparameters: learning rate (set constant), batch size and learning time (number of epochs).

The training objective of the model was to minimise the BCE loss L. To have a better understanding of the actual precision and recall of the trained models, the models were evaluated on the validation set using the percentage of detected joints (PDJ) metric (Toshev and Szegedy, 2014) on the detected centrepoints with a fraction of 0.1. Hereby, the torso diameter is defined as the diameter of the bounding box of the flower. Based on the PDJ score the precision, recall and F1 scores were calculated.

The model with the highest F1 score on the validation set was selected as the final model.

2.4. Flower localisation

To accurately adjust the harvester to the height of the flowers, the 3D location of the flowers needs to be determined. To verify the possibility of determining the 3D location using the above-mentioned flower detector, the flower field simulator is extended with a stereo vision system. By imaging the flowers now from two different viewpoints the 3D location of the flowers can be detected. This section expands on how the camera system can be integrated into the harvesting machine, the algorithms to determine the 3D position of the flowers and the validation in a virtual world of this process.

2.4.1. Stereo vision camera setup

A stereo vision camera system consists of two cameras which, by combining the image information of both cameras, makes it possible to extract depth information from objects that are perceived by both cameras. In this work, we propose the usage of a stereo vision system which consists of two industrial graded RGB cameras to determine the location of the flowers. For further calculations, we based the system on cameras with a sensor size of 1/2”, a focal length of 6 mm, and a resolution of 1280*1024 pixels, although these would be cropped to a resolution of 512*512 pixels to match the input of the detection network.

We positioned the two cameras at 1 m above ground level and 15 cm apart in the direction of travel. With this configuration, the resulting stereo system can capture a width of 54 cm at 45 cm above ground level, the average height of the Calendula flowers. In the direction of travel, however, the field of view is 26 cm, which is more narrow. This implies that a sufficiently high frame rate is required to capture every part of the field in the direction of travel. Since harvest will take place at a maximum of 3 km h^{-1} a frame rate of at least 4 fps is required. Figure 7 illustrates the described configuration of the stereo vision system.

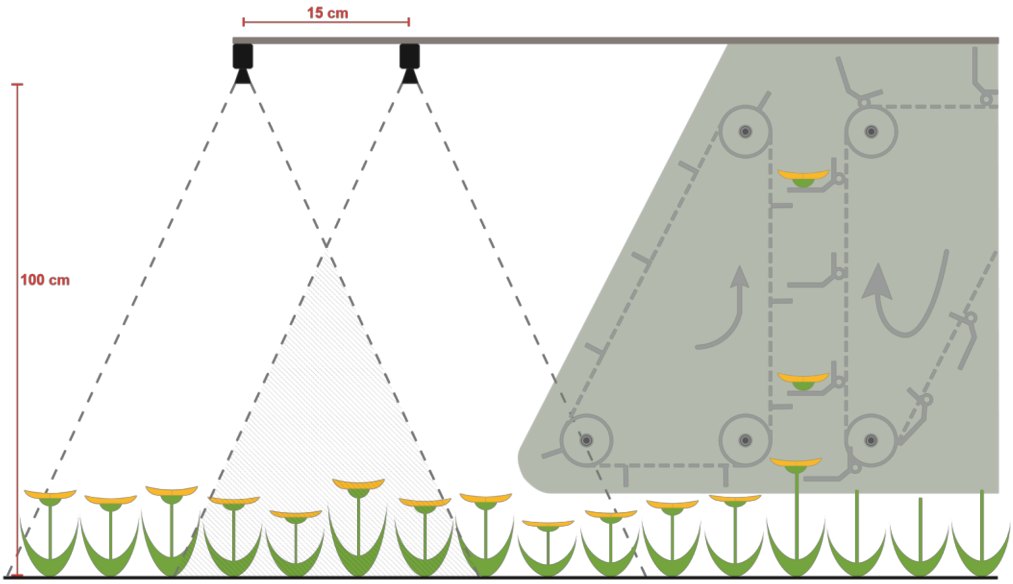


Figure 7: Side view of the stereo vision camera setup with the field of view of the stereo camera in the driving direction illustrated.

2.4.2. Localisation algorithm

The process of determining the 3D position of the flowers with stereo vision consists of several steps, as Fig. 8 illustrates. First of all, the flowers are detected in both the images from the left and the right camera. After detection, the pixel coordinates of the corresponding flowers in both images are matched. This results in a centrepoint in pixel coordinates for each flower in the field of view of the stereo vision system. Triangulation of these coordinates results in a 3D position of the flowers in the coordinate system of the first stereo vision camera, which can be transformed to a world coordinate system (Szeliski, 2011). This algorithm was implemented using OpenCV (version 4.5.3) functionality in Python (version 3.9.12).

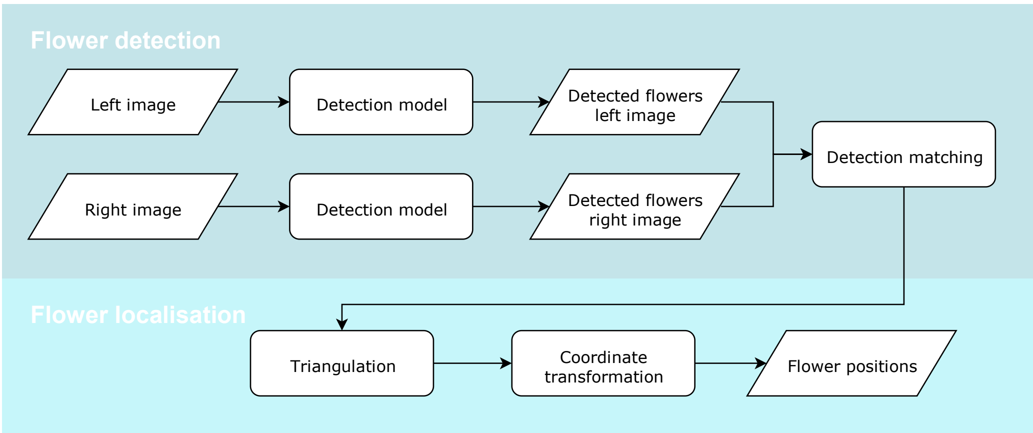


Figure 8: Detection and localisation of Calendula flowers using a stereo vision camera system.

2.4.3. Validation on a virtual flower field

To validate the stereo vision setup, the flower field simulator in Unity has been extended with an extra camera to enable stereo vision and a checkerboard to virtually calibrate the system. Both cameras were configured with a sensor size of 1/2”, a focal length of 6 mm, and a resolution of 512 by 512 pixels.

To approach the mechanisms and use of a physical stereo system, both the internal and external camera parameters are determined by calibration in the virtual world. By capturing images from a checkerboard in different positions and orientations, the intrinsic parameters (centrepoint and focal length) of every camera were determined. The external camera parameters of the system were determined similarly, these consist of the pose of the right camera in relation to the left camera and the transformation matrix from the left camera coordinate system to the world coordinate system. To detect the checkerboard pattern and determine the camera parameters, the functionality provided by OpenCV was used.

With a calibrated camera setup, flower fields are simulated and captured with both cameras after which the localisation algorithm was applied.

To validate the accuracy of the predicted flower height, the prediction accuracy was determined for three different simulated fields. One with the flowers on the measured average height and two in which the average height is in- or decreased with the standard deviation.

3. Results

The results of this work can be divided into three categories: data collection and generation, sim-to-real learning for flower detection, and the localisation of Calendula flowers with stereo vision. Each of these results is discussed in the following sections.

3.1. Data collection and generation

In this study, different datasets were collected and generated to create the proposed pipeline to generate synthetic agricultural data. Besides, a test set of real images was collected to evaluate the sim-to-real transition of the detection model. This section lists the results of the data collection and generation. All collected data, the 3D models of Calendula plants, and the generated synthetic dataset are published on Zenodo under CC-BY licence (Vierbergen et al., 2022).

3.1.1. Field data

Using the camera setup with an Intel RealSense D415 depth camera, 1954 images of a Calendula field were captured. By capturing images at various moments during the flowering season, at different plots, and under different weather and light conditions, the dataset holds a wide variety. Figure 9 and figure B.11 in Appendix B show a sample of these images.

Figure 9: Top: examples of real images from the captured dataset. Bottom: images generated with the proposed synthetic data pipeline.

The Calendula flowers are most clearly visible in the RGB and depth images when the camera is placed at 120 to 140 centimetres above the ground with a pitch angle between 0 and 20 degrees. More detail about the collected dataset can be found in Appendix A.

Table 1 lists the measured flower count, heights, and diameters of Calendulas in different plots. A high variety in flower height is observed both between different plots and within one plot. The measured Calendula flowers are positioned at an average height of 44.6 centimetres above the ground and have an average diameter of 5.98 cm.

Table 1: Number of flowers, average height and diameter with standard deviation of flowers in a sample of 1 m^2 in different plots.

Plot	Flowers	Height (cm)	Diameter (cm)
A	24	49.2 ± 6.7	4.0 ± 1.1
B	26	46.3 ± 5.5	4.2 ± 1.0
C	31	41.4 ± 4.2	3.8 ± 1.1
D	26	45.1 ± 5.0	4.1 ± 1.1
E	26	47.2 ± 5.8	4.6 ± 1.3
F	38	39.6 ± 4.9	6.2 ± 1.1
G	76	43.3 ± 4.0	6.4 ± 0.9
H	24	51.4 ± 3.8	6.1 ± 1.4

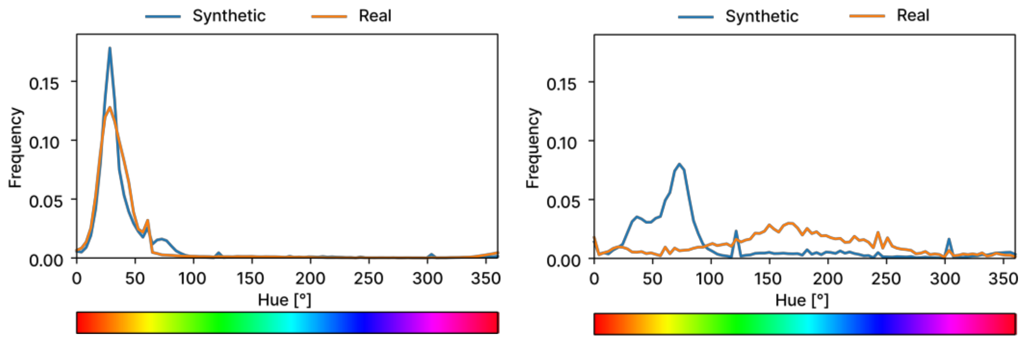
3.1.2. Photogrammetry

In total 980 images were taken in the controlled environment of a photo booth. This enabled the creation of 29 3D models of Calendula flowers and 7 different structures of leaves with the use of photogrammetry.

3.1.3. Synthetic dataset

In only a short amount of time, the flower field simulator is able to create a large dataset. On a laptop equipped with an Intel i7-8550U CPU and a Radeon Pro WX 3100 GPU it took us 20 minutes to create the training set of 15.000 synthetic images. Figure 9 shows a few of the generated images next to images of real Calendula flowers. The synthetic images can mostly be easily distinguished from the real ones by looking after collage effects in the background, unrealistic lighting, colour schema, and arrangement of the objects.

Despite clear differences, the synthetic images can be recognised as images of a Calendula field and clearly share some characteristics with the real images. In both, the same types of objects appear: flowers and leaves of Calendula plants, soil, weeds, and varied lighting conditions. The colour distribution of both datasets share similar characteristics as well. To quantify this, the images are converted to HSV colour space, and the distribution of the hue value is studied. Figure 10 shows that in both the real and synthetic datasets the hue of the flowers is very similar. In both, the mode is 28°. It is noticed however that for the other parts of the images, where no flowers occur, the distribution differs. The peak at 70° indicates the colour of the limited available leaf assets that were frequently used in generating the synthetic images without augmenting their colour. The background colour is thus in reality still more diverse than the background generated in the flower field simulator.



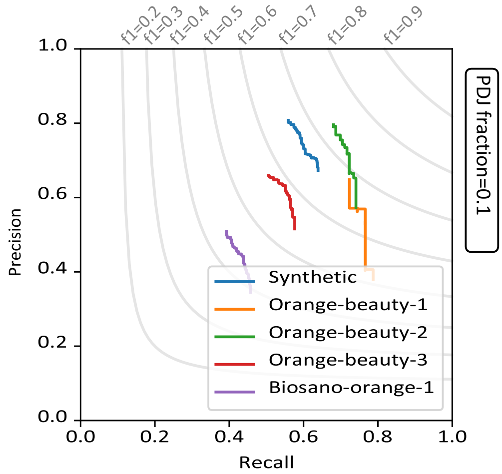
(a) Hue distribution of flowers

(b) Hue distribution of image background

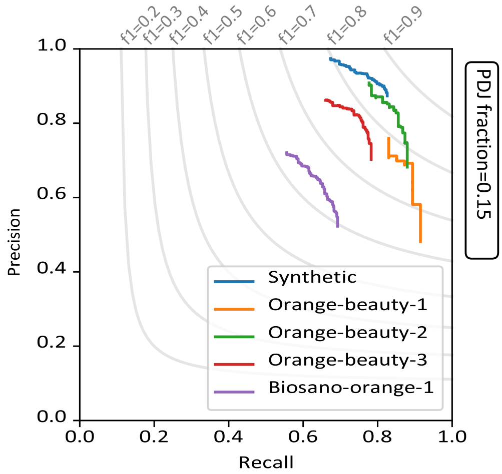
Figure 10: The colour distributions of real and synthetic images show similar characteristics.

3.2. Sim-to-real flower detection

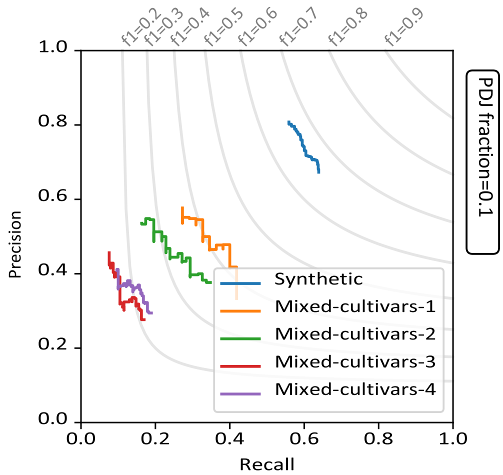
The final detection model was trained for six epochs with a batch size of eight and a learning rate of 1e-5. To make the sim-to-real transfer, this model was tested on nine test sets: one test set of synthetically generated data, four test sets of orange Calendula flowers (cultivars Orange Beauty and Biosano Orange), and four test sets that hold up to fifteen different cultivars. More details about the test sets are included in Appendix A and Appendix B. In order to match the resolution of real images captured with the RealSense to the input of the detection model, the images in the test sets were divided in two along their horizontal centre, after which both halves were cropped to a resolution of 512 by 512 pixels. Both cropped halves where then inputted to the network. Tested on the selected model, the precision, recall, and F1 score on these test sets are reported in Fig. 11. In this figure, the trade-off between recall and precision is made by both the threshold applied to the heatmap prediction of the detection model outputs and the set PDJ fraction. A smaller PDJ fraction challenges the detection model to detect the centre of the flower accurately, while a larger fraction allows some offset to the centre.



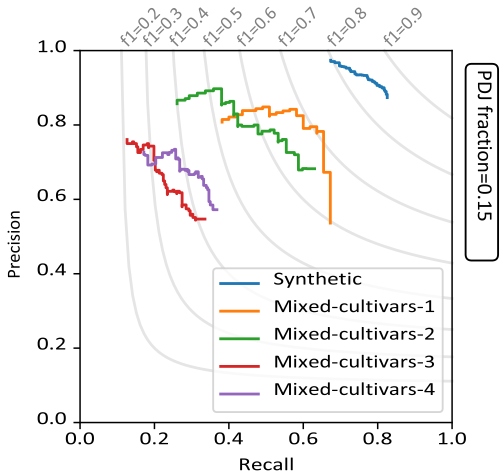
(a) Detection on orange flowers with a maximum PDJ fraction of 0.10.



(b) Detection on orange flowers with a maximum PDJ fraction of 0.15.



(a) Detection on flowers with mixed colour and a maximum PDJ fraction of 0.10.



(b) Detection on flowers with mixed colour and a maximum PDJ fraction of 0.15.

Figure 11: Sim-to-real transfer of the detection model on test sets of real images of Calendula flowers. Top: test sets with orange flowers. Bottom: Test sets with a diverse set of flower colours.

For the evaluation with the PDJ fraction, the centrepoint of a flower is defined as the centre of its bounding box. However, the true centre can deviate largely from this definition in case the flower is on the edge of the image. Because of this, the PDJ score is largely affected by flowers on the edges of the image. To mediate this, a border of 28 pixels in the 512 by 512 input image was created in which the detections are not taken into account for the evaluation on the test sets. Further, a change of perspective or a tilt of the flower can also result in a difference between the true centre of a flower and the centre of its bounding box. In this evaluation, there is no compensation made for these effects and its assumed that the centre of the bounding box is a good approximation of the true centre.

The sim-to-real transfer is made best on the test sets with orange flowers and a PDJ fraction of 0.15. In this case, the F1 score reaches up to 84%. By increasing the PDJ fraction up to 0.25, the F1 score increases to 86% on test set *Orange-beauty-2*. Since the average diameter of the measured Calendula flowers is 5.2 cm, a PDJ fraction of 0.10 and 0.15 respectively correspond with a maximum deviation of about 5.2 and 7.8 mm from the centrepoint of the flower in the horizontal plane. This is the case when the flower is not or is only slightly tilted.

Since the detection model is trained on images that simulate an Intel RealSense sensor at a height of 120 to 140 cm above ground, a loss in performance is observed when the sensor is set at a height of 82.5 cm in test set *Orange-beauty-1* compared to the F1 score on test set *Orange-beauty-2*. Since both test sets were captured at the same moment and of the same plants with only a difference in height of the RealSense sensor the decrease in F1 score can be assigned to the difference in sensor height.

The detection model is able to infer 24 frames per second. This provides the needed speed to capture every part of a flower field.

3.3. Flower localisation

With the use of a virtually created flower field, the localisation accuracy of the stereo vision setup is tested. A total of three different fields are generated to this end. In these fields, the Calendula flowers were positioned at a height of 38.6, 44.6 and 50.6 cm. Figure 12 shows one pair of the generated images with the detected and matched flower pairs annotated. Next, Fig. 13 shows the difference between the measured and true location of the flowers.

This shows that the stereo vision system can determine the height of the flowers with an average error of 6.9 ± 5.1 mm.

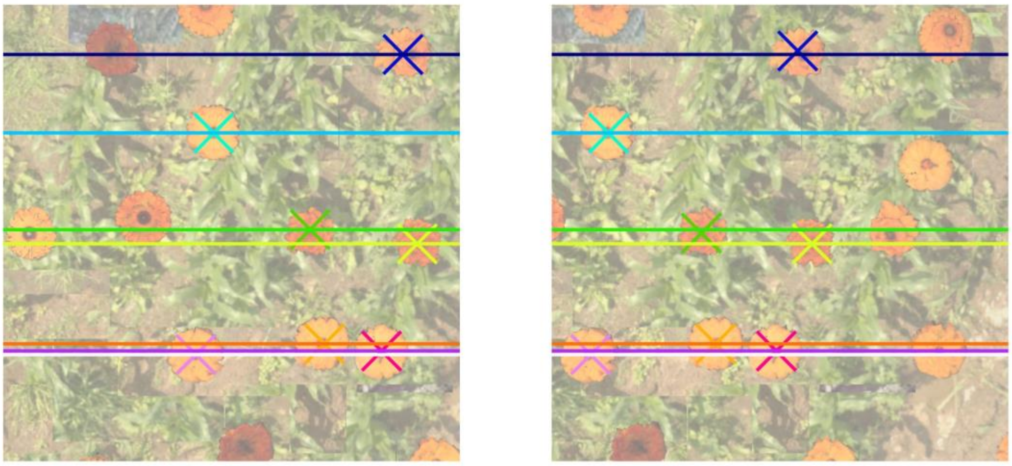


Figure 12: Left and right image of stereo vision system with detected and matched flowers annotated in corresponding colours, together with the epipolar lines. The images are made 50% transparent to increase the visibility of the annotations.

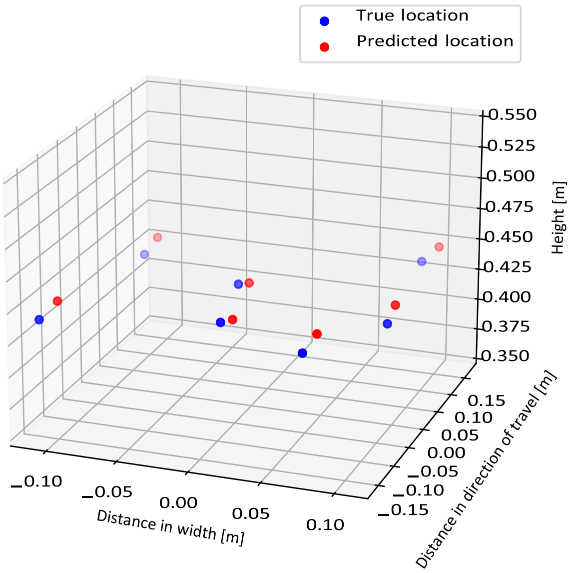


Figure 13: The difference between the true and predicted location of the flowers. The average error in height is 6.9 ± 5.1 mm.

4. Discussion

4.1. Synthetic data pipeline

The proposed simulation pipeline makes it possible to quickly generate a large amount of varied image data. This enables quick development iterations without having to wait for the next harvest, or season to collect more data on the crop. With this, we can mediate the need and costs of collecting and labelling a large dataset of real images. Our pipeline can generate thousands of varied synthetic images with a minimum number of 3D models used as input. Although the background colour in the synthetically generated images is dominated by the green colour of the leaf assets, the use of the proposed pipeline offers the possibility to add even more variation than what is easy to capture in the real world. We introduced, for example, diverse soil and weed types by collecting and combining image data available from the different datasets. This could be extended for the leaf colours as well.

The proposed pipeline and especially the flower field simulator show a big potential to create synthetic training data for agricultural applications. A limitation, however, in line with the remarks of Rizzardo et al. (2020), is that the flower field simulator based on Unity only renders field data. There is currently no integration with a robot operation system (ROS) to include robotic simulation or physical simulation of the flowers which would create an end-to-end virtual test platform for the harvest of Calendula flowers.

4.2. Sim-to-real flower detection

While only trained on synthetic images, the detection model is capable of detecting flowers in real-world images with an F1 score of up to 84% when a PDJ fraction of 0.15 is applied. This is a lower F1 score compared to other recent studies that utilise deep learning to detect flowers. For instance, Dias et al. (2018) reports an F1 score of up to 92% for the detection of apple flowers. In the recent work of Wang et al. (2022), the detection of pear flowers was demonstrated using synthetic data. Their best model achieved an F1 score of up to 96%. However, they used the synthetic data as a supplement to a dataset of real images and did not make the full sim-to-real transfer.

Although it is possible to compare F1 scores, it is still hard to make a good one-on-one comparison with the other works. This is because the targeted flower species, the used model, and the used metrics to determine true and false positives differ. In our study, for example, the use of a small PDJ fraction is an additional criterion in the evaluation compared to other studies.

Further, the test sets of most other studies are not made publicly available. This raises the barrier to evaluating our detection model and its sim-to-real capabilities in other conditions or for other flower species.

The sim-to-real capability of our model is highly influenced by the colour and cultivar of the flowers. For the detection model to perform well on a wider range of cultivars and flower colours, the domain randomisation in the flower field simulator has to be extended to include a wider variety of cultivars. This illustrates the potential of the proposed pipeline to generate synthetic data. With only a few example plants of the other cultivars, new 3D models can be generated and used as an asset in the flower field simulator to generate new synthetic data.

4.3. Flower localisation

The localisation of Calendula flowers shows to be possible with an average error of 6,9 mm in height. This should make it possible to integrate the localisation system on a Calendula harvester and automatically adjust the harvester to the height of the flowers.

A limitation of this work is that the localisation is only validated on a limited number of synthetic images. Although the results are promising, field tests have to be carried out in the future to validate the localisation on real data. A possible difficulty for the localisation on real data can be the matching of corresponding flowers between the left and right image. Much denser coverage of flowers, overlapping and tilted flowers are some of the complexities that will occur in an outdoor environment and were not present in the synthetic data on which the localisation was tested on.

Further, this work is not only relevant to the automated harvest of flowers but also to a wide range of precision agricultural applications. Other relevant applications for the developed sim-to-real pipeline and localisation system are yield prediction, weed management, fruit harvest, and variability mapping.

5. Conclusion

This work demonstrates how synthetically generated data can help accelerate the development of precision agricultural applications that require a huge amount of training data. To this end, we designed a simulation pipeline that makes use of photogrammetry and a flower field simulator to create synthetic images of a Calendula field. Secondly, we trained a flower detector on the synthetic images and demonstrated a successful transfer from simulation to reality. This transfer was validated on a large and diverse set of real Calendula images. Next, this detector has been used in combination with a stereo vision system to determine the positions of the flowers towards automating the harvest of Calendula flowers. As a final contribution, the collected and generated data for this study is published on Zenodo to further stimulate research on precision agriculture.

In future work, the flower detection and localisation systems should be implemented on a real Calendula harvester. With this implementation, the effect of a dynamic height adjustment on harvest efficiency can be studied.

The relevance of this work is however much broader than the harvest of Calendula flowers. The proposed synthetic data pipeline is flexible and can be adapted to simulate other crops and agricultural processes. Leveraging the potential of sim-to-real learning can eliminate costs and can accelerate the development of precision agricultural applications.

Acknowledgements

The authors would wish to thank everyone who contributed to this work in a direct or indirect manner. Special thanks go towards Aaron Van Gehuchten for the technical assistance in creating the 3D models and Peter De Roovere for the insightful discussions about CenterNet and its loss functions.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Metadata

Table A.1: Characteristics of collected data series

Data serie	Cultivar	Condition	Camera		Number of images		Test set
			Height (cm)	Pitch (°)	RGB-D	Annotated	
1	Orange Beauty	C1	82.5	10	58	85	Orange-beauty-1
2	Orange Beauty	C1	120	10	78	78	Orange-beauty-2
3	Orange Beauty	C2	140	15	95	0	n/a
4	Orange Beauty	C2	120	50	200	0	n/a
5	Orange Beauty	C2	110	50	280	0	n/a
6	Orange Beauty	C2	140	50	204	0	n/a
7	Biosano Orange	C3	140	20	558	100	Biosano-orange-1
8	Orange Beauty	C3	140	20	414	100	Orange-beauty-3
9	Mixed Cultivars [2]	C4	140	10	15	15	Mixed-cultivars-1
10	Mixed Cultivars [2]	C5	140	10	15	15	Mixed-cultivars-2
11	Unknown mix	C6	140	10	5	5	Mixed-cultivars-3
12	Unknown mix	C7	140	10	5	5	Mixed-cultivars-4

[1] See Table A.2.
[2] Cultivars in data series: 15001, 15537, 2997 109/112, Biosano Orange, Erfurter Orange, Nova, Red With Black Center, Ringelblume, Corniche d'Dor, Yellow Gem, Orange Beauty Vreeken, Lemon Beauty, Carola, Apricot Beauty en 2008294

Table A.2: Time, location and weather conditions of data series.

Condition	Date (dd/mm/yyyy)	Time	Location	Weather
C1	17/09/2020	14:21	Merelbeke, Belgium	Clear, sunny
C2	05/11/2020	14:09	Molenbeek-Wersbeek, Belgium	Clear, sunny [1]
C3	09/07/2021	13:19	Letterhoutem, Belgium	Cloudy
C4	09/08/2021	10:24	Merelbeke, Belgium	Overcast
C5	09/08/2021	15:04	Merelbeke, Belgium	Clear, sunny
C6	09/08/2021	15:24	Merelbeke, Belgium	Clear, sunny
C7	09/08/2021	16:58	Merelbeke, Belgium	Clear, sunny

[1] Frost during night before.

Appendix B. Dataset examples

(a) Orange-beauty-1 (b) Orange-beauty-2

(c) Biosano-orange-1 (d) Orange-beauty-3

(e) Mixed-cultivars-1 (f) Mixed-cultivars-2

(g) Mixed-cultivars-3 (h) Mixed-cultivars-4

(i) Synthetic

Figure B.1: Samples of the different test sets used to validate the sim-to-real transfer.

References

Andújar, D., Calle, M., Fernández-Quintanilla, C., Ribeiro, A., and Dorado, J. (2018). Three-Dimensional Modeling of Weed Plants Using Low-Cost Photogrammetry. *Sensors*, 18(4):1077. https://doi.org/10.3390/s18041077

Bechar, A. and Vigneault, C. (2016). Agricultural robots for field operations: Concepts and components. *Biosystems Engineering*, 149:94–111. https://doi.org/10.1016/j.biosystemseng.2016.06.014

Borkman, S., Crespi, A., Dhakad, S., Ganguly, S., Hogins, J., Jhang, Y.-C., Kamalzadeh, M., Li, B., Leal, S., Parisi, P., and others (2021). Unity Perception: Generate Synthetic Data for Computer Vision. *arXiv preprint arXiv:2107.04259*.

Cieslak, M., Khan, N., Ferraro, P., Soolanayakanahally, R., Robinson, S. J., Parkin, I., McQuillan, I., and Prusinkiewicz, P. (2022). L-system models for image-based phenomics: case studies of maize and canola. *in silico Plants*, 4(1):diab039. https://doi.org/10.1093/insilicoplants/diab039

de Melo, C. M., Torralba, A., Guibas, L., DiCarlo, J., Chellappa, R., and Hodgins, J. (2022). Next-generation deep learning based on simulators and synthetic data. *Trends in Cognitive Sciences*, 26(2):174–187. https://doi.org/10.1016/j.tics.2021.11.008

Deng, J., Dong, W., Socher, R., Li, L.-J., Kai Li, and Li Fei-Fei (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, Miami, FL. IEEE. https://doi.org/10.1109/CVPR.2009.5206848

Dias, P. A., Tabb, A., and Medeiros, H. (2018). Apple flower detection using deep convolutional networks. *Computers in Industry*, 99:17–28. https://doi.org/10.1016/j.compind.2018.03.010

Dwibedi, D., Misra, I., and Hebert, M. (2017). Cut, Paste and Learn: Surprisingly Easy Synthesis for Instance Detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1310–1319, Venice. IEEE. https://doi.org/10.1109/ICCV.2017.146

Hasan, A. S. M. M., Sohel, F., Diepeveen, D., Laga, H., and Jones, M. G. (2021). A survey of deep learning techniques for weed detection from images. *Computers and Electronics in Agriculture*, 184:106067. https://doi.org/10.1016/j.compag.2021.106067

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Kamilaris, A. and Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147:70–90. https://doi.org/10.1016/j.compag.2018.02.016

Khalid, K. A. (2012). Biology of Calendula officinalis Linn.: Focus on Pharmacology, Biological Activities and Agronomic Practices. *Medicinal and Aromatic Plant Science and Biotechnology*, page 16.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollar, P. (2020). Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327. https://doi.org/10.1109/TPAMI.2018.2858826

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dolla´r, P. (2015). Microsoft COCO: Common Objects in Context. Number: arXiv:1405.0312 arXiv:1405.0312 [cs].

Lu, Y. and Young, S. (2020). A survey of public datasets for computer vision tasks in precision agriculture. *Computers and Electronics in Agriculture*, 178:105760. https://doi.org/10.1016/j.compag.2020.105760

Mavridou, E., Vrochidou, E., Papakostas, G. A., Pachidis, T., and Kaburlasos, V. G. (2019). Machine Vision Systems in Precision Agriculture for Crop Farming. *Journal of Imaging*, 5(12):89. https://doi.org/10.3390/jimaging5120089

Nikolenko, S. I. (2021). Synthetic data for deep learning (Vol. 174). Springer Nature.

Olsen, A., Konovalov, D. A., Philippa, B., Ridd, P., Wood, J. C., Johns, J., Banks, W., Girgenti, B., Kenny, O., Whinney, J., Calvert, B., Azghadi, M. R., and White, R. D. (2019). DeepWeeds: A Multiclass Weed Species Image Dataset for Deep Learning. *Scientific Reports*, 9(1). https://doi.org/10.1038/s41598-018-38343-3

Picon, A., San-Emeterio, M. G., Bereciartua-Perez, A., Klukas, C., Eggers, T., and Navarra-Mestre, R. (2022). Deep learning-based segmentation of multiple species of weeds and corn crop using synthetic and real image datasets. *Computers and Electronics in Agriculture*, 194:106719. https://doi.org/10.1016/j.compag.2022.106719

Prakash, A., Boochoon, S., Brophy, M., Acuna, D., Cameracci, E., State, G., Shapira, O., and Birchfield, S. (2019). Structured Domain Randomization: Bridging the Reality Gap by Context-Aware Synthetic Data. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7249–7255, Montreal, QC, Canada. IEEE. https://doi.org/10.1109/ICRA.2019.8794443

Qiu, W. and Yuille, A. (2016). UnrealCV: Connecting Computer Vision to Unreal Engine. In Hua, G. and Jégou, H., editors, *Computer Vision – ECCV 2016 Workshops*, volume 9915, pages 909–916. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

Rahnemoonfar, M. and Sheppard, C. (2017). Deep Count: Fruit Counting Based on Deep Simulated Learning. *Sensors*, 17(4):905. https://doi.org/10.3390/s17040905

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, Las Vegas, NV, USA. IEEE. https://doi.org/10.1109/CVPR.2016.91

Rizzardo, C., Katyara, S., Fernandes, M., and Chen, F. (2020). The Importance and the Limitations of Sim2Real for Robotic Manipulation in Precision Agriculture. Number: arXiv:2008.03983 arXiv:2008.03983 [cs].

Roh, Y., Heo, G., and Whang, S. E. (2021). A Survey on Data Collection for Machine Learning: A Big Data - AI Integration Perspective. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1328–1347. https://doi.org/10.1109/TKDE.2019.2946162

Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., and McCool, C. (2016). DeepFruits: A Fruit Detection System Using Deep Neural Networks. *Sensors*, 16(8):1222. https://doi.org/10.3390/s16081222

Szeliski, R. (2011). *Computer Vision*. Texts in Computer Science. Springer London, London.

Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., and Abbeel, P. (2017). Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. *arXiv:1703.06907 [cs]*. arXiv: 1703.06907.

Toshev, A. and Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660.

Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., Cameracci, E., Boochoon, S., and Birchfield, S. (2018). Training Deep Networks with Synthetic Data: Bridging the Reality Gap by Domain Randomization. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1082–10828, Salt Lake City, UT. IEEE. https://doi.org/10.1109/CVPRW.2018.00143

Veselinov, B., Adamovic, D., Martinov, M., Viskovic, M., Golub, M., and Bojic, S. (2014). Mechanized harvesting and primary processing of Calendula officinalis L. inflorescences. *Spanish Journal of Agricultural Research*, 12(2):329. https://doi.org/10.5424/sjar/2014122-4876

Vierbergen, W., Willekens, A., Dekeyser, D., Cool, S., and wyffels, F. (2022). Sim2real Flower Detection Towards Automated Calendula Harvesting. Type: dataset. https://doi.org/10.5281/zenodo.6945367

Wang, C., Wang, Y., Liu, S., Lin, G., He, P., Zhang, Z., and Zhou, Y. (2022). Study on Pear Flowers Detection Performance of YOLO-PEFL Model Trained With Synthetic Target Images. *Frontiers in Plant Science*, 13:911473. https://doi.org/10.3389/fpls.2022.911473

Wang, R., Zheng, Z., Lu, X., Gao, L., Jiang, D., and Zhang, Z. (2021). Design, simulation and test of roller comb type Chrysanthemum (Dendranthema morifolium Ramat) picking machine. *Computers and Electronics in Agriculture*, 187:106295. https://doi.org/10.1016/j.compag.2021.106295

Willoughby, R. A., Solie, J. B., Whitney, R. W., Maness, N. O., & Buser, M. D. (2000). A mechanical harvester for marigold flowers. In Proceedings. ASAE Annu Int Meeting, Milwaukee, WI, USA (pp. 1-14).

Zhang, Q., Liu, Y., Gong, C., Chen, Y., and Yu, H. (2020). Applications of Deep Learning for Dense Scenes Analysis in Agriculture: A Review. *Sensors*, 20(5):1520. https://doi.org/10.3390/s20051520

Zhou, X., Wang, D., and Krähenbühl, P. (2019). Objects as Points. Number: arXiv:1904.07850 arXiv:1904.07850 [cs].