Instance-based CycleGAN for object segmentation with few annotations

D. N. Díaz Estrada[1], O. Robert[1], M. Kresovic[1], C. Torres[1], D. Muselet[2], and A. Tremeau[2]

[1] Vilmorin-Mikado, Limagrain Group, France
[2] Université Jean Monnet Saint-Etienne, CNRS, Institut d Optique Graduate School, Laboratoire Hubert Curien UMR 5516, F-42023, SAINT-ETIENNE, France
{daviddiaz997}@gmail.com

Abstract. Nowadays deep networks provide excellent results in the context of object segmentation. Available models have been trained on common objects and are not designed to segment specific objects such as fruits or vegetables. In order to help breeders to accelerate and to modernize the process of agriculture products phenotyping, it is necessary to fine tune general models on specific species. Nevertheless, a minimum amount of annotations are required for this retraining step. In this paper, we propose a solution to minimize the annotation workload for each specie. The main idea consists in leveraging the annotations of one specie A in order to fine tune a model on a specie B with few annotations. For this purpose, we propose an Instance-based CycleGAN (ICG) that creates synthetic images of specie B along with corresponding annotations. By fine tuning a segmentation network with these synthetic images and annotations, we show that this network can obtain very good performance on the new specie B, without requiring to manually annotate a large amount of images for this specific specie B.

Keywords: CycleGAN, Instance segmentation, Unsupervised, Synthetic annotations, Plant phenotyping

1 Introduction

Proper detection and localization of agriculture products, such as fruits and vegetables, are critical for the growth of the economy in the Agri-food sector [1]. Particularly fruit detection is essential in agriculture products phenotyping due to the benefits it brings to crop breeding activities. Digital, automatic, and reliable phenotyping technologies are highly important to boost the advancement of genetic gain in breeding programs [2]. In fact, Computer Vision, Deep Learning, and Image-based Technologies applied to plants phenotyping aid to reduce the workload of scientific researchers by automatically measuring phenotypic indicators which accelerates the progress toward crop breeding optimization [3, 2]. Moreover, plant phenotyping involves challenging computer vision tasks that require advanced image segmentation models that distinguish between diverse instances of objects in an image [2]. In recent years seeds companies started to develop Deep Learning models for precise fruit detection and segmentation of images acquired under uncontrolled conditions for the purpose of phenotyping. However, the annotation process required to build training datasets constitutes a bottleneck considering the change in varieties and typologies of species every year.

Since experts have already spent time to annotate images of few species, in this paper, we are trying to leverage this annotation to automatically annotate images of new species. Considering that we have a good amount of images and annotations for one specie, called A, and only few images and annotations for a second specie B, we propose a solution to create additional synthetic images of specie B along with accurate mask annotations, as illustrated on Fig. 1. And we show that this additional data can clearly boost the segmentation results on specie B.

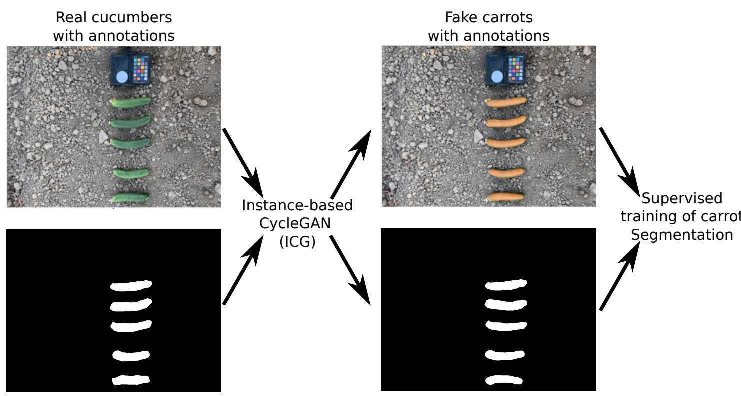

Fig. 1. Our Instance-based CycleGAN is able to independently transform each instance from one specie to another one while preserving the background. Thus, starting from a labeled dataset of cucumbers (specie A) and few carrot images (specie B), we can create many fake carrot images with accurate synthetic masks to train a new network.

Generative adversarial networks [4] are good solutions to create realistic synthetic images of a new specie, and specially to solve the problem of image-to-image translation [5, 6]. In this case, it is possible to provide images from two species and learn to translate from one specie to another. While Pix2Pix requires paired images where the objects have to be located at the same position in the images [5], CycleGAN can be trained with unpaired data thanks to a cycle consistency process [6]. This network is more adapted to real applications where we have images of two different species acquired under uncontrolled conditions. Nevertheless, the classical training of a CycleGAN can't solve our problem of missing annotations for the second specie B. Indeed, Fig. 2 illustrates the image translation process when using a classical CycleGAN transforming a cucumber (specie A) image to a carrot (specie B) image. The network is able to provide realistic carrot images but without any mask annotations for these fake images.

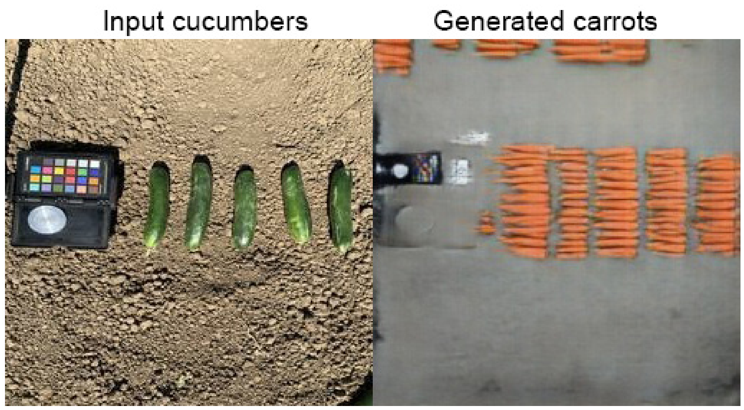

Fig. 2. Example of CycleGAN failed experiments.

In order to solve this problem, we propose a new approach called Instance-based CycleGAN (denoted ICG, hereafter) that preserves the background of the input image (specie A) and transforms individually and independently each instance of this image to the second specie B. With this solution, we can control the image location where we paste our new fake object and hence accurately deduce the mask annotations of each instance of specie B (see. Fig.1).

The main contributions of this work are the following:

- We show that CyceGAN is a good solution to create synthetic images in the context of fruits/vegetables segmentation.
- We propose an Instance-based version of CycleGAN, that is able to transform independently each instance of one specie to a second specie.
- We run extensive experiments on real data showing that our solution helps in segmenting carrots in real images acquired under uncontrolled conditions.
- We fine tune and test the classical Mask R-CNN [7] on our synthetic images.

The rest of the paper is organized as follows: the related works are presented in section 2 while section 3 is devoted to our proposed CycleGAN pipeline. The experiments and results are described in section 4 and finaly, we conclude and propose future works in section 5.

2 Related works

Generative Adversarial Networks (GANs) GANs were introduced in [4], and since their appearance in the field of Deep Learning they have achieved outstanding results in many domains, such as image colorization [8], super-resolution [9], image inpainting [10], and image-to-image translation [5, 6]. Conditional GANs (CGANs) were proposed to guide the data generation process by conditioning it to a given input variable [11]. In fact, CGANs were employed for Image-To-Image translation tasks, where the generation of output images is conditioned to a given input image, as in Pix2Pix [5] and CycleGAN [6]. While Pix2Pix works with paired training data, CycleGAN handles unpaired image-to-image translation, which gives much wider possibilities to experiment with translation models between domains and scenarios where paired correspondences are simply not available, such as when transforming zebras to horses, apples to oranges, or cucumbers to carrots. Given that there are no pair correspondences in CycleGAN, the generation process is regulated by incorporating a cycle consistency constraint in the translation, which is inspired by the idea that when translating from one domain to another, and then back to the original domain, the initial information must be preserved [6].

It has been reported that the cycle consistency loss in CycleGAN does not necessarily ensure structural consistency between input and generated image [12], which can be problematic in the case of data generation for segmentation purposes since the corresponding segmentation mask of the input image (from domain A) would not match the structure of the generated image (from domain B). In order to solve this, some approaches have been proposed in the literature, which include: SECLEGAN [13], CySGAN [14], and the model proposed by Yang et al. [12]. However, these previous approaches [13, 14, 12] do not change the shape of objects (object transfiguration) during translation, since they impose structural constraints to ensure consistency between input and generated images. In contrast, InstaGAN [15] does allow object transfiguration but it increases the complexity of the network by including a context preserving loss.

Our method is different since we leverage the capacity of CycleGAN to change the shape of instances during translation without increasing the complexity of the network. Furthermore, our method ensures that the network sees single instances of each domain during training, and thus is able to learn features on the instance level with higher details, which enhances the realism of the generated instances and allows to work with domains that highly differ in color, texture, and shape, and therefore to perform unpaired instance-to-instance translation. Moreover, we will show in section 4 that by working on an instance level, the network generates instances of another domain with a clean background, which allows the estimation of the corresponding groundtruth segmentation mask.

Segmentation Mask R-CNN was proposed in [7], it extends Faster R-CNN [16] by including an additional output branch to predict segmentation masks, in parallel with the existing branch for classification and bounding box regression. In our research work, we use the Detectron2 [17] framework developed by Facebook, which is a state-of-the-art library for object detection and segmentation algorithms; it includes different architectures such as Faster R-CNN, Mask R-CNN, RetinaNet [18], Panoptic FPN [19], and DensePose [20]. Recently, Detectron2 has been used in the agriculture sector in [21], where the authors worked on instances segmentation of olive trees, soil, and shadow areas.

3 Instance-based CycleGAN

Our main task consists in creating synthetic images of carrots with mask annotations, so that we are able to train a carrot segmentation network. Assuming that we have a large image dataset of cucumbers with mask annotations and a small dataset of carrots with mask annotations, we propose to supplement the carrot dataset with fake images that are transformed versions of cucumber images. For this purpose, we train an instance-based CycleGAN (ICG) from a subset of the cucumber dataset and the full set of real carrot dataset. The idea is to train our ICG with balanced data between cucumbers and carrots. This is illustrated in Fig. 3, where we can see that the cucumber dataset is split in two subsets Cc1 and Cc2. Our ICG is trained with the set Cc1 and the real carrots RC1. Then, the trained ICG is used for inference on the Cc1 subset to create the fake carrots FC1, while the subset Cc2 is giving birth to the fake carrots FC2. These two sets of fake carrots (FC1 and FC2) are considered differently in the experiments, since FC1 is obtained from Cc1 which has been used for training ICG while FC2 is obtained from completely new cucumber images Cc2, not used for ICG training.

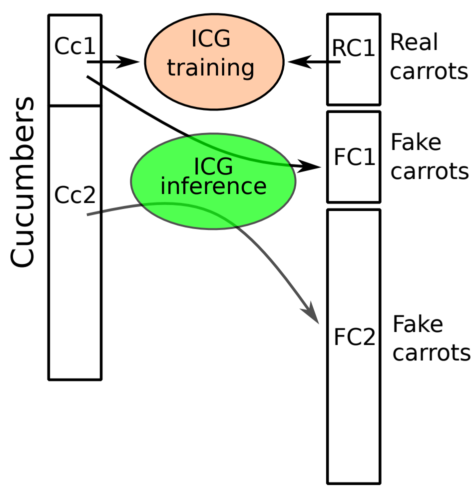

Fig. 3. The exploitation and creation of the cucumber and carrot images. All the cucumber images are real images with mask annotations.

The training step of our ICG is detailed in Fig. 4. The inputs are from both side cucumbers (Cc1) and carrots (RC1) along with the corresponding mask annotations. The provided masks are used to extract each instance of the images and to paste it in a single image with black background. Since the size of both datasets Cc1 and RC1 are low, we apply classical geometrical data augmentation to each single image before feeding the network. Then, we apply the classical CycleGAN training process with cycle consistency and unpaired instance images [6].

To be precise, let consider the species (*domains*) A and B (which are cucumbers and carrots, in our illustration). We have two generators G_A and G_B creating fake images of specie A and B, respectively, and two discriminators D_A and D_B which are trained to differentiate true from fake images of specie A and B, respectively. The complete loss we are minimizing while training our ICG is the following:

$$\begin{aligned} \mathcal{L}(G_A, G_B, D_A, D_B) = & \mathcal{L}_{GAN}(G_A, D_A, A, B) \\ & + \mathcal{L}_{GAN}(G_B, D_B, B, A) \\ & + \lambda_{cyc} \mathcal{L}_{cyc}(G_A, G_B) \\ & + \lambda_{id} \mathcal{L}_{id}(G_A, G_B), \end{aligned} \tag{1}$$

where the GAN losses $\mathcal{L}_{GAN}$ are the classical adversarial losses that control that the transforms from A to B and from B to A are correct. The cycle loss $\mathcal{L}_{cyc}$ controls that the model can transform one specie to another and back ($A \to B \to A$ and $B \to A \to B$). And the identity loss $\mathcal{L}_{id}$ regularizes the generators to be near identity mappings when real samples of the target domain are provided as the input to the generator. This regularization has been shown to improve color composition in the mappings. λ_{cyc} and λ_{id} control the balance between the losses.

The inference stage is detailed in Fig. 5. For this step we just consider the transform from cucumber to carrot, and not the inverse since our aim is to supplement the carrot dataset. Thus, the inputs are a cucumber image and the corresponding mask image. The first step consists in the extraction of the instances from the image and to paste it on a small image with black background. Then our trained ICG is used to transform each cucumber instance into a fake carrot. A simple thresholding on the result image can provide the corresponding mask annotation. Finally, given the input original image and annotations, the created fake carrots and their masks, we can paste the fake carrot on top of the original cucumber image and provide the corresponding mask annotation. Note

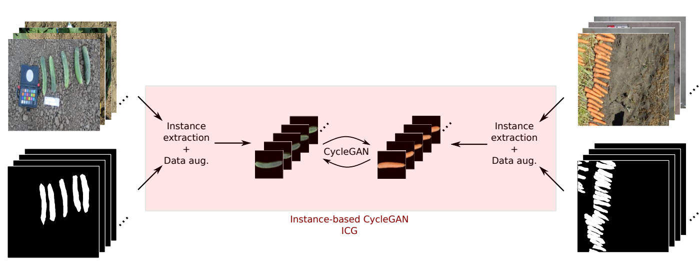

Fig. 4. Training step of our Instance-based CycleGAN.

that the masks for the fake carrots are different from the masks of the input cucumbers since ICG is free to change the shape of the vegetables, depending on the shapes of the trained images (cucumbers and carrots).

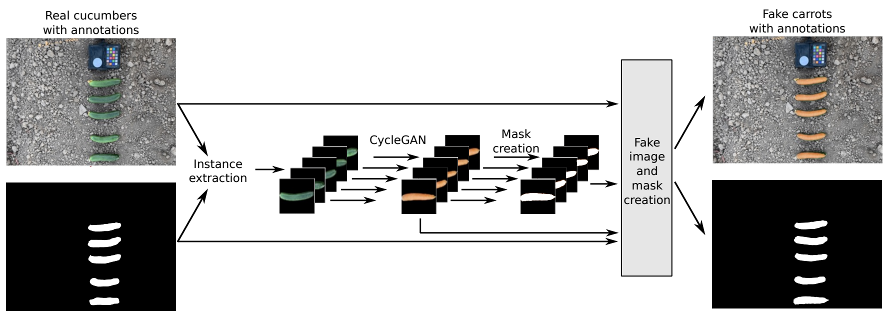

Fig. 5. Inference step of our Instance-based CycleGAN.

The fake carrots and their annotations are used to train a segmentation network in a supervised way.

3.1 Implementation

The images used in all the experiments were captured by breeders and varietal selection technicians. For the CycleGAN experiments, we used the CycleGAN Tensorflow 2 implementation available in [22]. Similarly as in [6], in all our experiments we used a batch size of 1, a learning rate of 0.0002 with a linear decay after the first 100 epochs, and a maximum total of 200 epochs. The weights of cycle consistency and identity losses were set to $\lambda_{cyc} = \lambda_{id} = 10$ (eq. 1) for all the experiments.

For the segmentation experiments, we used Detectron2 [17] framework version "v0.6" and Mask R-CNN with feature pyramid network (FPN) built with Resnet-101 as the backbone for instance segmentation (RESNETXT-101-FPN-3X). We used this architecture in all our segmentation experiments. For the transfer learning we used the pre-trained weights on ImageNet [23] and then retrain the network for carrot segmentation. All segmentation models were trained for 1000 epochs, with a learning rate of 0.0025, and the evaluation period was set to every 100 epochs, where the evaluation during training is done with the test images. Moreover, during inference and test, we used a threshold of 0.5 as the minimum value for a prediction to be considered positive.

4 Experiments

4.1 Instance-based CycleGAN

Creating fake carrots. Since our objective is to use a small amount of annotated data, we selected only 265 annotated carrots (RC1 dataset from Fig. 3) and 273 annotated cucumbers (Cc1 dataset from Fig. 3). We applied the preprocessing steps of our CycleGAN pipeline, and in this case, 3 random rotations were applied to each original image from each domain in order to achieve at least 1000 training examples per domain.

Figure 6 show examples of our CycleGAN pipeline, illustrating that it is possible to generate realistic carrots, with variety in texture and color. The shape is also modified during CycleGAN translation and the segmentation annotations of the fake carrots are estimated correctly. Moreover, the results of inference on cucumbers not seen during training are satisfactory, which shows the robustness of CycleGAN inference on new data.

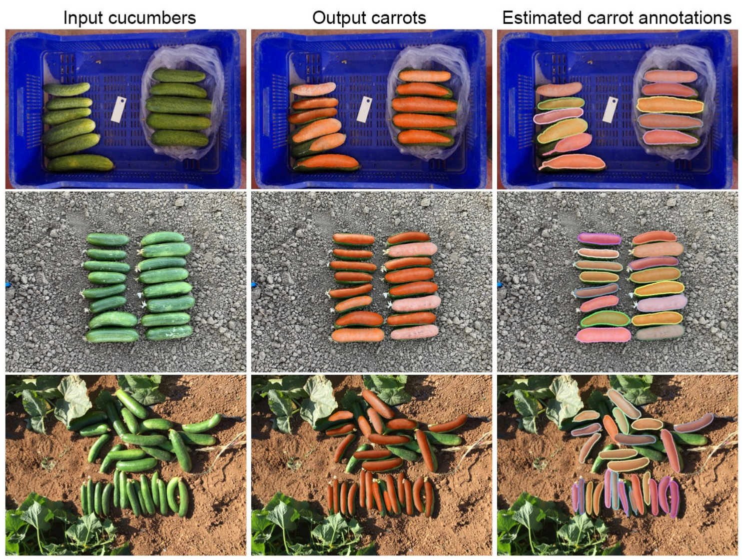

Fig. 6. Illustration of the creation of fake carrot images.

Squashes and peppers. In order to show that this approach can be applied to many different vegetables, we also train an ICG between peppers and squashes. For this ICG, the difference between the shapes of the two species is more important than between carrots and cucumbers. We train the ICG with 133 original pepper and 240 original squash annotations, with random rotations to reach 1200 training images on each domain.

Figure 7 shows examples of transform between the two species. As we can see our ICG is able to generate fake squashes and fake peppers which are realistic in shape even with variations in texture and color. Figure 8 shows examples for the full pipeline, creating images with several fake instances and their corresponding annotations. As we can see, the instance segmentation annotations are estimated correctly in both cases.

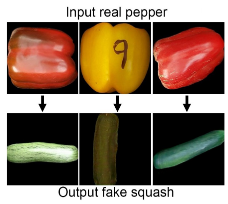

Fig. 7. Examples of Squash/Pepper instance transforms.

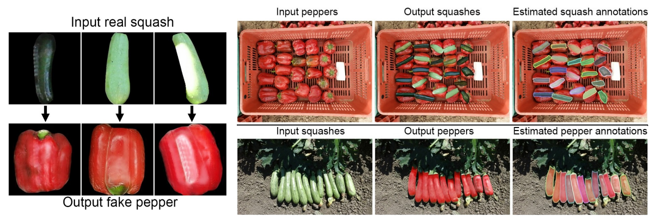

Fig. 8. Examples of Squash/Pepper images and annotations transform.

4.2 Segmentation

For the segmentation experiments, we used real and fake carrot data (RC1, FC1 and FC2 presented in Fig. 3) for training the segmentation network. For testing, we have created a dataset of 1928 annotated carrots, acquired with different cameras under uncontrolled environments, with diversity in illumination conditions and backgrounds. Moreover, the carrot images are also challenging for a segmentation task regarding the number of instances per image, with a minimum of 52, a maximum of 218, and an average of 122 carrots per image.

Evaluation Metrics We measured the performance of the segmentation models with the following metrics: precision, recall, IoU [24], F1 score [25, 26], and Average precision (AP). For COCO metrics [27], AP is the average across multiple IoU levels (the minimum IoU value to consider a prediction as positive), in fact, AP is measured as the average AP for IoU from 0.5 to 0.95 with a step size of 0.05. Moreover, AP_{50} and AP_{75} are measured at IoU of 0.5 and 0.75, respectively. In our research context, we use the COCO AP metric included in Detectron2. In this work, the main idea is to create synthetic data to train a network and not to provide realistic images. Consequently, we don't assess the realism and quality of the generated images.

Results. Table 1 shows the results of the evaluation on the real carrot images (1928 annotated carrots). Note that these images are from real carrots (no fake image) and have never been seen during the segmentation network training. They are real unseen test images. We provide the results on these images for three different sets of training data:

- Training on RC1 : 265 annotated carrots, the ones used for training ICG.
- Training on RC1 + FC1. FC1 is constituted by 273 fake carrots, outputs of our ICG by transforming the cucumbers Cc1, which have been used to train ICG.
- Training on FC2 : FC2 is constituted of 513 fake carrots, outputs of our ICG by transforming the cucumbers Cc2, which have not been used to train ICG.

The results in Table 1 clearly show the improvements provided by our approach. Indeed, by using only the few real annotated data (RC1) to train the segmentation network, we get an Average Precision (AP) of 70.52. If we supplement this training set with fake carrots provided by our ICG, the AP increases

Table 1. Segmentation results on real carrot images. Note that the 273 fake carrots with an "*" are the transformations of the 273 cucumbers used to train our ICG, while the 513 carrots of the last row are the transformations of 513 cucumbers not seen during the ICG training.

Training data	**AP**	**AP$_{50}$**	**AP$_{75}$**	**Precision**	**Recall**	**IoU**	**F1**
Original carrots RC1 (265)	70.52	86.32	83.87	94.90	89.01	84.80	91.66
Original carrots RC1 (265) + Fake carrots FC1 (273)	72.46	86.50	84.17	**95.44**	90.09	86.39	92.58
Fake carrots FC2 (513)	**74.56**	**89.29**	**88.07**	93.79	**92.18**	**86.88**	**92.93**

to 72.46, showing that fake carrots can be used to train a segmentation network and improve the results on real carrots. Finally, if we consider only fake carrots (FC2) for training the network, we outperform the two previous results and get 74.56 as AP. This shows that fake data is not only useful to supplement a small training set, but can be used to train a network without requiring any real data.

Consequently, we can conclude that our ICG has two main advantages:

- **Annotation reduction.** The experiments show that we can train a CycleGAN with only around 250 annotations from each domain and use real and fake images to improve the results over a network that has been trained only on real images. So without new annotations, the accuracy is boosted by adding fake data.
- **Robustness.** The experiments show that our ICG can be fed with new cucumbers (not seen during the ICG training) to provide fake carrots with annotations that can be used to train a segmentation network that outperforms all the other networks trained with real data. This shows that ICG can generalize to new images.

5 Conclusion

We have developed a CycleGAN pipeline that allows the network to perform unpaired instance-to-instance translation and create a synthetic dataset for instance segmentation. Thanks to this work we are able to leverage annotated data from one domain and use it for the benefit of a second domain with less available data. Furthermore, we created a carrot segmentation model of images acquired under uncontrolled conditions and increased the segmentation accuracy by including fake data during training.

In addition, we show that our ICG is able to generate realistic instances, with diversity in color, texture, and shape, both for cucumbers/carrots and peppers/squashes dataset.

During our experiments, we have pasted each fake instance on the same background and similar location than the input real image, but it's clear that every new created fake instance could have been pasted on any new background with geometric augmentation applied to the object and the corresponding annotation.

This would open the door to infinite number of training data. This is the aim of our future works.

References

1. Lei Li, Qin Zhang, and Danfeng Huang, "A review of imaging techniques for plant phenotyping," *Sensors*, vol. 14, no. 11, pp. 20078–20111, 2014.
2. Zhenbo Li, Ruohao Guo, Meng Li, Yaru Chen, and Guangyao Li, "A review of computer vision technologies for plant phenotyping," *Computers and Electronics in Agriculture*, vol. 176, pp. 105672, 2020.
3. M. Fernanda Dreccer, Gemma Molero, Carolina Rivera-Amado, Carus John-Bejai, and Zoe Wilson, "Yielding to the image: How phenotyping reproductive growth can assist crop improvement and production," *Plant Science*, vol. 282, pp. 73–82, 2019, The 4th International Plant Phenotyping Symposium.
4. Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, Eds. 2014, vol. 27, Curran Associates, Inc.
5. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5967–5976.
6. Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2242–2251.
7. Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask r-cnn," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.
8. Kamyar Nazeri, Eric Ng, and Mehran Ebrahimi, "Image colorization using generative adversarial networks," in *Articulated Motion and Deformable Objects*, Francisco José Perales and Josef Kittler, Eds., Cham, 2018, pp. 85–94, Springer International Publishing.
9. Adrian Bulat and Georgios Tzimiropoulos, "Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
10. Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas Huang, "Free-form image inpainting with gated convolution," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4470–4479.
11. Hamed Alqahtani, Manolya Kavakli-Thorne, and Gulshan Kumar, "Applications of generative adversarial networks (gans): An updated review," *Archives of Computational Methods in Engineering*, vol. 28, no. 2, pp. 525–552, Mar 2021.
12. Heran Yang, Jian Sun, Aaron Carass, Can Zhao, Junghoon Lee, Zongben Xu, and Jerry Prince, "Unpaired brain mr-to-ct synthesis using a structure-constrained cyclegan," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Danail Stoyanov, Zeike Taylor, Gustavo Carneiro, Tanveer Syeda-Mahmood, Anne Martel, Lena Maier-Hein, João Manuel R.S. Tavares, Andrew Bradley, João Paulo Papa, Vasileios Belagiannis, Jacinto C. Nascimento, Zhi Lu, Sailesh Conjeti, Mehdi Moradi, Hayit Greenspan, and Anant Madabhushi, Eds., Cham, 2018, pp. 174–182, Springer International Publishing.

13. Vince Chen, Jin Wu, and Jiayi Luo, "Group2 seclegan: Improvement of the cyclegan with segmentation," Tech. rep., University of California San Diego, 2021, http://noiselab.ucsd.edu/ECE228-2021/projects/report/2Report.pdf, visited 2022-11-1.
14. Leander Lauenburg, Zudi Lin, Ruihan Zhang, Márcia dos Santos, Siyu Huang, Ignacio Arganda-Carreras, Edward S. Boyden, Hanspeter Pfister, and Donglai Wei, "Instance segmentation of unlabeled modalities via cyclic segmentation gan," 2022.
15. Sangwoo Mo, Minsu Cho, and Jinwoo Shin, "Instagan: Instance-aware image-to-image translation," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. 2019, OpenReview.net.
16. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, Cambridge, MA, USA, 2015, NIPS'15, p. 91–99, MIT Press.
17. Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick, "Detectron2," https://github.com/facebookresearch/detectron2, 2019.
18. Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
19. Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollar, "Panoptic feature pyramid networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
20. Riza Alp Güler, Natalia Neverova, and Iasonas Kokkinos, "Densepose: Dense human pose estimation in the wild," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7297–7306.
21. Asma Ben Abdallah, Abdelaziz Kallel, Mouna Dammak, and Ahmed Ben Ali, "Olive tree and shadow instance segmentation based on detectron2," in *2022 6th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, 2022, pp. 1–5.
22. Zhenliang He, "Cyclegan-tensorflow-2," https://github.com/LynnHo/CycleGAN-Tensorflow-2, 2017.
23. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.
24. Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 658–666.
25. Davide Chicco and Giuseppe Jurman, "The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, pp. 6, Jan. 2020.
26. Yutaka Sasaki, "The truth of the f-measure," *Teach Tutor Mater*, 01 2007.
27. Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár, "Microsoft coco: Common objects in context," 2014.