Data Augmentation for Leaf Segmentation and Counting Tasks in Rosette Plants

Dmitry Kuznichov, Alon Zvirin, Yaron Honen and Ron Kimmel
Computer Science Department, Technion IIT, Haifa 32000, Israel

Abstract

Deep learning techniques involving image processing and data analysis are constantly evolving. Many domains adapt these techniques for object segmentation, instantiation and classification. Recently, agricultural industries adopted those techniques in order to bring automation to farmers around the globe. One analysis procedure required for automatic visual inspection in this domain is leaf count and segmentation. Collecting labeled data from field crops and greenhouses is a complicated task due to the large variety of crops, growth seasons, climate changes, phenotype diversity, and more, especially, when specific learning tasks require a large amount of labeled data for training. Data augmentation for training deep neural networks is well established, examples include data synthesis, using generative semi-synthetic models, and applying various kinds of transformations. In this paper we propose a data augmentation method that preserves the geometric structure of the data objects, thus keeping the physical appearance of the data-set as close as possible to imaged plants in real agricultural scenes. The proposed method provides state of the art results when applied to the standard benchmark in the field, namely, the ongoing Leaf Segmentation Challenge *hosted by* Computer Vision Problems in Plant Phenotyping.

1. Introduction

Visual context, scene understanding, and object location seem to be key factors in image augmentation for deep neural networks. There are many ways to augment data in images. One of the most prominent ways is cutting objects from the original image, and pasting the objects, exercising geometrical transformations, into a synthetic image. Often these operations lead to non-realistic or even non-logical output. Gould *et al*. overcome this problem by understanding the image scene [18]. Dvornik *et al*. find the importance of object locations in the original images and use these characteristics when deploying the object onto the synthetic image [10, 11].

Several papers dealing with plant phenotyping convey the importance of data augmentation. One reason is that training deep neural networks requires a large ground-truth data-set, which is not always available. Even if such a data-set exists, augmentation serves to vary the training set, thus improving the learning procedure and performance. Recent surveys on plant phenotyping emphasize the need for data augmentation, and transfer learning in the sense that synthetic data can and should be used for training networks, later tested on real data [21]. Main considerations include sufficient amount of balanced data, annotation and normalization of data, and outlier rejection [41]. Synthetic data modeling, graphical rendering, and transfer learning in context of using pre-trained deep networks (or at least their first layers) play a key role in plant genotyping and phenotyping [9].

Data augmentation and synthesizing images is gaining acceptance and practice. The *KITTI* and *Cityscapes* datasets are used extensively for semantic understanding of urban scenes [12]. Basic practices include rotations, cropping, color transforms; advanced methods are usually applied to specific domains. For example, Richardson *et al*. synthesized human facial models by learning parametric geometric and texture features. [34, 35, 40]. Integrating parametric surface modeling with a *Generative Adversarial Network* for generation of realistic human face textures is suggested by [42].

Although applied deep learning is common in analysis of plant structure, and computational and heuristic graphical modeling techniques exist, few attempts have been suggested to combine them. Leaf counting and instance segmentation remain a challenge, due to diverse leaf shapes, size and variability during their life cycles in the growth stage, and also due to overlapping and occlusions, abundant number of different crops, and diverse real-world environments (laboratory, greenhouse, field).

Here, we propose a method integrating both approaches, by presenting a method of data augmentation preserving the photorealistic appearance of plant leaves, and using the augmented data as training set for a network architecture known to achieve high quality results in instance counting and segmentation, *Mask R-CNN* [19]. We focus on augmenting a plant image training set with photorealistic syn-

thetic images. Using a limited amount of images of real leaves and accurate manual segmentation, we use geometric transformations and image processing tools to create a practically infinite amount of synthetic images simulating real-life environments. Among these, some manipulations can be considered global, like rotations and scaling, while some are tailored specifically for a particular dataset, for example, number of leaves and their orientations in a plant, following a set of formal rules supplemented by random parameter distribution in a reasonable range.

The *Computer Vision Problems in Plant Phenotyping* (CVPPP) dataset was created specifically for expected contributions in image based learning related to plant phenotyping [38, 4, 26]. The rosette image dataset is complemented by two ongoing competitions, the *Leaf Segmentation Challenge* and *Leaf Counting Challenge* (LSC, LCC respectively), hosted and maintained by CVPPP [25]. Arabidopsis was selected as it is the plant with best known genetics, has a short life span, and a dataset was created in a controlled environment with manual annotations of leaf masks as ground truth. Several approaches tackling this dataset are described in [39]. Introduced in 2014, the dataset and ongoing challenge already gained considerable impact in plant phenotyping research [45]. We also tested our methods on another plant image dataset, collected by Rahan-Meristem [32], as part of a pilot phenotyping project, for future research into early detection of plant stress and prediction of growth stages. This set consists of 50 images of mature avocado in a plantation, with accurate manual segmentation of all leaves. Each of these images contain between 20 and 80 leaves.

We propose two methods for augmenting an image set by generation of photorealistic synthetic images, preserving geometry and texture as appearing in complex real world agricultural scenes. We demonstrate the applicability of these methods to boost a deep neural network performance in accurately counting and segmenting leaves in diverse photographing conditions. Our main contribution is simulation of data to enlarge the existing data-set with a novel method of synthesizing realistic plant images. In the next section we review several papers concerned with data augmentation aimed specifically for identifying plant parts, especially rosette leaves. The *Methods* section describes our approach and strategy for collaging leaf images as means for data augmentation, and the *Results* section presents qualitative and quantitative results.

2. Related Efforts

Taking a deeper inspection at recent efforts focusing on data augmentation by synthesizing leaf images, most draw their ideas from three main approaches: *Graphical Modeling*, *Domain Randomization* and *Generative Adversarial Networks*. Other attempts addressing leaf segmentation and counting rely heavily on neural networks (aimed at image processing tasks), but use a limited augmentation, or train on other datasets, or apply pre-processing, such as color transform, brightness and contrast adjustments, but not specifically designed for fine contouring of leaf shape nor refined realistic texture. Ubbens & Stavness [47] introduce an open source platform for plant phenotyping, provide pre-trained networks for common plant phenotyping tasks.

Graphical Modeling. Formalizing plant structure by mathematical models was introduced by Lindenmayer, known as L-systems. Formal grammars with a set of rules (functions) are utilized to produce *chains* of elements representing plant parts - stems, leaves, roots. These models originated in an attempt to assist biological understanding of cell structure and development by formal mathematical models [23]. Later, these ideas were applied in graphical simulation of plants [31], for rendering synthetic images, and for creating augmented datasets required to train deep neural networks. Mundermann *et al*. empirically model 3D graphical representations of arabidopsis [27]. After collecting thousands of measurements of real plants from seedlings to maturity, they infer growth curves of shape, size and position of leaves and stems, and their development over time. Ubbens *et al*. introduced a parametric version of L-systems for generating synthetic rosettes [46]. Simulating growth stages by parametrizing plant components, they argue that images of real/synthetic plants are significantly interchangeable when training a neural network.

Domain Randomization. The main purpose of *Domain Randomization* is to tackle the task of object localization, instance detection and possibly object segmentation. A few works demonstrate the capability of training entirely on synthesized images, intended for testing on real world scenes. This approach intentionally abandons photorealism by random perturbations of the environment, such as random textures, thus attempting to force the neural network to learn the essential features of the objects [44]. In practice, this is implemented by developing a simulator producing randomized rendered images. The reasoning is that with enough variability in the simulator, images from the real world should appear to the model as just another variation [43]. Applying *Domain Randomization* to arabidopsis images is described by Ward *et al*. [48]. Their method synthesizes random textures of leaves and background, and constructs separate leaves by deforming a canonical template of a leaf. Leaf positions are randomized in a unit sphere, and random camera positions and lighting are applied to produce the images. The main drawbacks of this approach are that leaves are assumed to be planar, textures have a cartoon-like appearance, and it does not handle background.

Generative Adversarial Networks. Recently achieved

popularity for creating realistic image sets, Generative Adversarial Networks (GAN), first introduced by Goodfellow *et al*. [17], are intended for training neural networks. Two recent papers apply *conditional Generative Adversarial Networks* (cGAN) to generate artificial images of arabidopsis plants, targeting the Leaf Counting/Segmentation challenge. The condition serves as restriction on the training process, is supplemented to the input, and fed to one or more of the networks' layers. Giuffrida *et al*. propose a process starting with random noise, taking number of leaves as the condition concatenated through the networks' layers, ending up with 128×128 pixel RGB images of simulated arabidopsis [16]. Zhu *et al*. first produce masks in a structured manner, used as input to a GAN [50]. Individual leaf masks are selected from the ground truth masks, split into 5 folders by size, and arranged in a logical order by rotations and zooming with small randomization, and placing smaller leaves on top. These synthesized mask images serve as condition to the generator, which outputs pseudo-real images, replacing leaf masks and mask background with RGB texture.

Limited Augmentation. In addition to the previously mentioned articles, all incorporating data augmentation and synthetic image generation, several other approaches have been applied for segmentation and counting tasks, in particular dealing with the arabidopsis dataset. Pape & Klukas used image processing tools - colorspace transform, Gaussian blur, morphological operators and Euclidean distance maps to distinguish between individual leaves, and trained a random forest classifier for leaf border detection [29, 30]. De Brabandere *et al*. propose a discriminative loss function for clustering pixels belonging to the same instance and use flipping, rotations and scaling for augmenting the training set [7].

Several attempts using Recurrent Neural Networks (RNN) have been proposed: Romera-Paredes & Torr [36] suggested an architecture starting with a CNN to extract image features. Ren & Zemmel performed dynamic *Non Maximal Suppression* to handle occlusions [33]. Salvador *et al*. added a encoder-decoder model [37]. A recent RNN based article also reports on collection of a new arabidopsis dataset, time-series image sequences of four accessions under controlled acquisition, in hope and expectation of further research [28].

Other efforts tackle only the counting problem, and treat counting as a direct regression problem, without attempting to segment individual leafs. Dobrescu *et al*. use a modified version of Resnet50, applying limited augmentation - rotations, zooming, and flipping of the original images [8]. Aich & Stavness apply intensity saturation, Gaussian blur and corresponding sharpening, in addition to rotations and flipping, and use a modified version of SegNet [1]. Counting leaves by learning features in a non-supervised dictionary learning fashion, without neural networks, was considered by [15]. Giuffrida *et al*. designed a deep learning architecture for leaf counting, using augmentation during training [14]. A note should be made that efforts aimed at counting, even if best at predicting number of objects in a snapshot, do not directly address leaf texture nor geometry. Other attempts, while performing fine positioning of leaves [46, 48, 50] generate or rely on exaggerated synthetic leaves, lack real-world textures and diverse geometry, especially leaf contours and their appearance in digital images. The reader interested in works concerned with image analysis of rosette plants, which do not mention data augmentation nor synthetic generation, is also referred to [20, 2, 49, 22, 3, 5]. In the next section, we discuss data augmentation for the specific task at hand, by image synthesis using a model in which leaves are extracted from images of real plants.

3. Methods

Presented here is a method for generation of synthetic images, a technique we term *collage*. The basic idea is creating a set of segmented leaf images on a transparent background, a single leaf per image, using manual annotations or an automatic procedure. In the basic scheme, single leaves undergo geometric transformations with random parameters in a fixed range, and pasted in random locations over selected backgrounds. The advanced scheme takes into account the logical-semantic relationships among objects, in this case structuring and positioning of leaves as part of the whole plant. We apply an algorithm specifically tailored for generating images that seem highly realistic, as if actually taken from the target dataset. In case of plants, especially if photoed during a controlled environment, the structuring of leaves as part of the whole plant is important, as well as collaging a photorealistic image in terms of geometry, texture, occlusions, and background.

The basic and advanced schemes are termed *naïve collage* and *structured collage*, respectively. The naïve collage is intended for "in the wild conditions", when minimal collection of data and annotations are available, and still expecting to do some predictions. The structured collage exploits certain plant structural attributes assumed or known to be correlated to measurable phenomena. We address the following issues, and elaborate on them in the following paragraphs: leaf (object) shape, size, location, ordering and positioning of leaves (object parts) as part of the whole plant, and image background.

3.1. Naïve Collage

The naïve collage is composed of previously segmented objects on a transparent background. The objects are then positioned on selected background images, AS IS, without any logical-semantic relationships among objects. At first,

this technique was tried on a relatively basic scenario: The raw data consisted of 50 high resolution RGB images of avocado leaves (3000×4000 pixels), supplemented with high quality manual annotations (mask per object) of most visible leaves, see Figure 1.

Figure 1: Mature avocado leaves: Left – original image, Center – original with manual annotations, Right – annotation detail. Marked leaves are alpha blended for display purposes.

From the original set of annotated avocado images we extract a set of "suitable" leaves, by three criteria: (1) Size (2) Not occluded by other objects (3) Clear and focused appearance. The resulting set consists of $\sim$200 leaves (out of $\sim$3000 original leaves), a sample displayed in Figure 2. The reasoning behind discarding occluded leaves is so the generated image will be as realistic as possible; in the wild it is uncommon to see cut leaves, and the appearance of partial objects is a side effect of the collage algorithm. Small and blurry looking leaves are removed due to our deliberate intention of detecting only leaves having a clear and sharp appearance in the image. These masked leaves were scaled to 600 pixels in the largest dimension, preserving the aspect ratio, thus enabling to collage a few dozen leaves in a single 1024×1024 image such that each leaf can be seen clearly.

Figure 2: Examples of segmented avocado leaves used for naïve collages.

As background for the synthesized images we used cropped images of size 1024×1024 from a set of 24 high resolution agricultural images, not including clearly seen avocado leaves (see Figure 3). The rationale for background selection is based on the intention of accurately detecting fine looking objects, (i.e., leaves of a certain crop), and distinguish them from other botanical objects (stems, fruit, ground), mostly appearing in images as a composition of green shades.

Figure 3: Examples of agricultural scenes used as background for naïve collages.

The collage is created by positioning 10 to 40 segmented leaves (random number in arbitrary but fixed range) in random locations, scale and rotations on top of a background image. The location of each leaf is randomly selected, the only restriction is that the leaf center remains inside the background image (1024×1024). Horizontal and vertical scaling are independent, with random values between 0.4 and 1.1. The rotation angle of each leaf (with respect to its original orientation) is randomly selected in the range $0-359$ degrees. We did not apply affine or projective transformations, in order to preserve the original point of view, although it can be done as well. Parallel to image creation, we generate corresponding masks fitting the created image; examples are displayed in Figure 4.

Figure 4: Generated images and corresponding masks from the avocado training set.

3.2. Structured Collage

The extended collage version takes into account the logical order, structure, and hierarchical relationship among objects placed in the image, specifically the location, size and shape of individual leaves as part of the whole plant. This is especially important in case of specific datasets such as the arabidopsis and tobacco images, all photoed from above, capturing plant structure at progressive development stages. It should be noted that although we present a description for synthesizing images with appearance akin to this specific dataset, similar steps, with fine tuning of parameters, can be employed for datasets based on different plant species or other image acquisition systems. In short, the process

for collage generation consists of selecting an appropriate background, creating a set of aligned leaves in canonical form, and logical insertion of leaves from this set onto the image. The workflow is depicted in Figure 5; details and considerations of this collaging process are described in the following paragraphs.

Background images. As a first step we created 112 background images, using original images from the dataset and applying a semi-manual segmentation of the plant from the background by activation of a heal-selection filter [13]. In total, $16, 26, 26$ and 44 background images were created, matching the A1, A2, A3 and A4 subsets, respectively. Examples are displayed in Figure 6.

Leaves in canonical form. In the next step all leaves were cut from the training set images according to their masks, and rotated to align them in canonical form. The rotation angle was determined as the angle between the horizontal axis and the leaf's principal axis, defined as a segment connecting the plant center (as manually marked) and the farthest mask pixel from the center. In all the aligned images the central pixel in the bottom row corresponds to the plant center. An example of original leaf image, principal axis and aligned form are depicted in Figure 7.

As a result 11096 aligned leaf images were created: $2088, 287, 146$, and 8575 in the A1, A2, A3 and A4 subsets, respectively. However, many of these leaves are not suitable for generation of realistic images; the main criteria for retaining leaves are: (1) the leaf mask contains no more than one component, (2) the leaf base is not too far from the plant center, and (3) the leaf appears fully (or almost) in the image, with minimal occlusion.

Examples of discarded leaves are shown in Figure 8. After removal, 5883 ($\sim 50\%$) were left: $1363, 219, 102$ and 4199 in the A1, A2, A3 and A4 categories, respectively. Note that although these leaves are discarded from the aligned set used in the generation procedure, similar looking partial and occluded leaves are expected to be detected and masked. The generation procedure, described in the next section, produces just this kind of occlusions, as well as full leaves.

Synthetic image generation. Following our intention to create realistic images, we heuristically attempt to imitate the plant's structure, using visual observations as rules of thumb: (1) The plant spawns from a point near the pot's center. (2) All the leaves grow from this point towards the periphery. (3) Leaf size and distance from the center are correlated. (4) Angles between leaves follow a certain distribution.

The main idea of image generation is the same for all datasets although each of the four subsets (A1-A4) should have its set of parameters fine-tuned. These include plant center, leaf size distribution and inter-leaves angle randomization. We define a *Length Mapping Vector* for each image in the training set, comprised of leaf lengths, sorted in descending order. These vectors induce the number of leaves to be inserted in the synthesized images, their ordering in the plant and their approximate sizes. The synthesized images are generated according to the following steps:

1. Randomly select a background image from the background image list of the specified set.
2. Randomly select the plant center coordinates, up to a maximal value from the image center.
3. Randomly select a *length mapping vector*.
4. Add first leaf (from the aligned set) with a random rotation angle. The leaf is selected from a subset of leaves with length in a range close (± 3 pixels) to the first value in the length mapping vector.
5. For each additional leaf:
   (a) Select leaf from the aligned set, with length approximating the corresponding length in the length mapping vector.
   (b) Select leaf angle based on previously added leaves. (last added leaf on top.)

In a formalized manner, let us define:
l_j as leaf j in a n-leaf dataset $L = \{l_j\}_{j=1}^{n}$,
I_k as background image k from the background set,
I_k^i as background image k with i added leaves, (note $I_k = I_k^0$, by definition),
and $T_i(I, l)$ as an operator adding leaf l to image I as i^{th} leaf in the image. (note $i \geq 1$).

The synthesized image is initialized as I_k^0. Adding leaf j to background image k containing $i-1$ previously added leaves results in image I_k^i, specified by

$$I_k^i \leftarrow T_i(I_k^{i-1}, l_j). \tag{1}$$

Since the process is iterative, it follows that,

$$I_k^i \leftarrow T_i(T_{i-1}(I_k^{i-2}, l_{j'}), l_j) \tag{2}$$

$$I_k^i \leftarrow T_i(\cdots T_1(I_k^0, l_{j_1}) \cdots, l_j). \tag{3}$$

The operator T_i is dependant on a few parameters, namely leaf location and rotation. T_1 is initialized with a plant center location (x, y), a first leaf angle α_1, and a length mapping vector $\vec{v}$. Values of x, y, α_1 are randomly chosen from a fixed range; $\vec{v}$ is selected from the mapping vectors of the dataset. These values remain constant till the end of a single image creation,

$$T_1 \Leftarrow ((x, y), \alpha_1, \vec{v}). \tag{4}$$

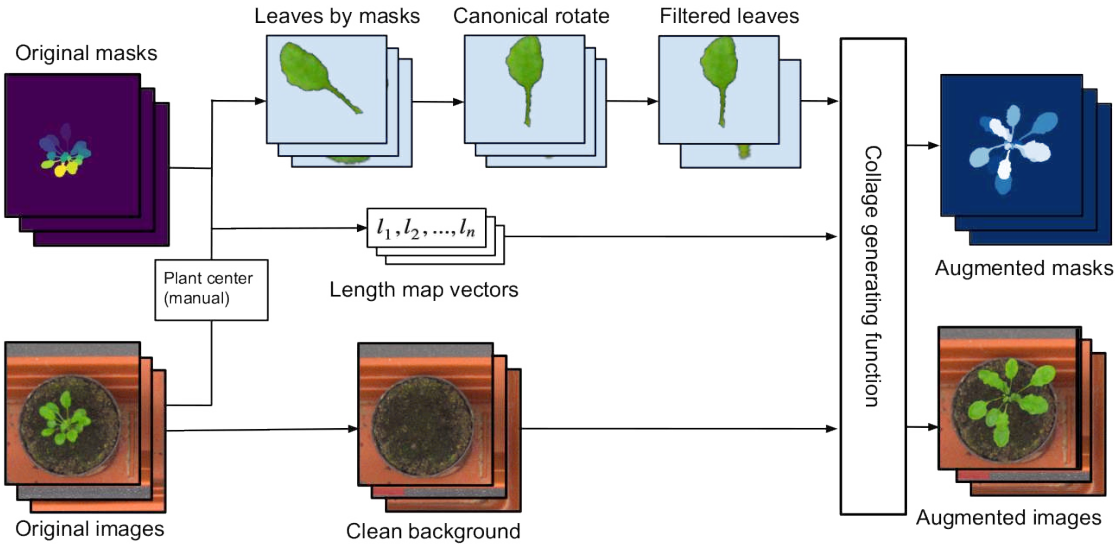

Figure 5: Structured collage generation pipeline.

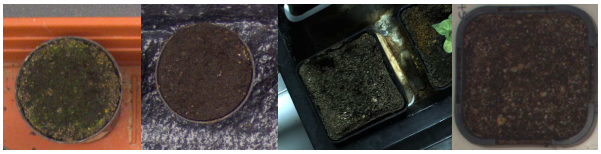

Figure 6: Clean background images, extracted from the A1-A4 rosette subsets.

Figure 7: Arabidopsis leaf: Left – original, Center – with principal axis originating in plant center, Right – aligned in canonical form.

Figure 8: Examples of arabidopsis leaves discarded from the generation procedure.

Obviously, while trying to simulate plant structure by object collaging, T_i's angle α_i is a function of number of objects and all angles of previously added objects, and can be defined iteratively by

$$\alpha_i = f(\alpha_{i-1}, i). \tag{5}$$

Taking a look at the rosette dataset [25], we notice approximately 120° between consecutive leaves, and similar to [46], a basic formulation of α_i can be stated as

$$\alpha_i = \alpha_{i-1} + 125^\circ \pm 10^\circ \tag{6}$$

Observing that rosette leaves grow in triads, with slight modification, first triad as before $\alpha_i = \alpha_{i-1} + 125^\circ \pm 10^\circ$, and first leaf (only) of each new triad $60^\circ \pm 10^\circ$, $30^\circ \pm 5^\circ$, etc.

Parameters chosen for each dataset are presented in Table 1. The size of the training images is restricted to multiples of 64 due to memory alignment. Plant center locations are from a threshold range surrounding the image center. Examples of synthesized images of rosette plants simulating the A1-A3 subsets are displayed in Figure 9.

Table 1: Datasets train parameters

Data-set	Original image size	Train image size	Image center	Plant center delta
A1	530×500	512×512	(256, 256)	40×40
A2	530×565	512×512	(256, 256)	40×40
A3	2448×2048	2048×2048	(1024, 1024)	160×160
A4	441×441	448×448	(224, 224)	35×35

4. Results

Segmentation and counting tasks were jointly performed with the publicly available Matterport implementation [24]

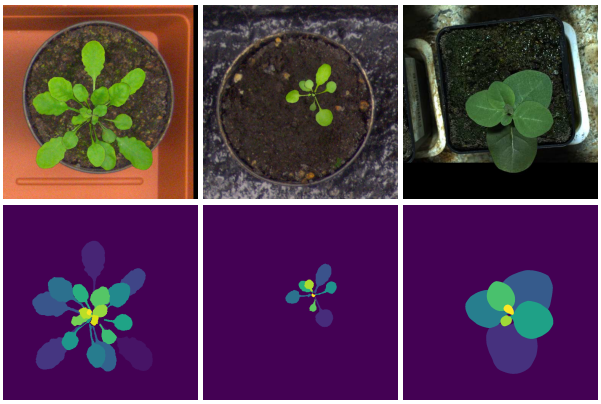

Figure 9: Examples of generated images and masks simulating the A1, A2 (arabidopsis) and A3 (tobacco) subsets.

of Mask-R-CNN [19], pre-trained on the COCO dataset. The naïve collage is used in the avocado case, more suitable for "in the wild" circumstances, containing images from various camera positions and include a variety of light conditions. The structured collage is aimed at the CVPPP dataset, acquired under controlled, consistent conditions, and exhibiting coherent plant structure.

4.1. Naïve Collage

To test the networks performance we use a single image from the manually annotated dataset, leaves of which were separated from the training set. These leaves are expected to be segmented by the network, after training on collaged images. We decided to evaluate correct segmentation by comparing leaf area for all leaves in the test image having mask areas over a fixed threshold (700 pixels). For correct segmentation we assume at least 0.8 IoU (Intersection over Union) of leaf area with respect to its manually annotated counterpart - mask area of the ground truth. Instance segmentation is visualized in Figure 10. Figure 11 shows number of leaves detected and misdetected, by training epochs.

4.2. Structured Collage

As detailed in the *Methods* part, the network was trained on structured collage images. Although the collages are generated from extracted leaves and corresponding masks, we did not use the original images, nor transformations of the originals, in the training process. For validation we used the original training data of the four subsets A1 - A4, containing ground truth instance segmentations per leaf. We did not use the leaf centers, nor the foreground/background masks of the whole plant, also supplied as part of the ground truth training data.

Figure 10: Avocado test image: original and segmentation result. Detected leaves are alpha blended for visualization; gray color indicates less than 0.8 IoU of leaf area with respect to the ground truth.

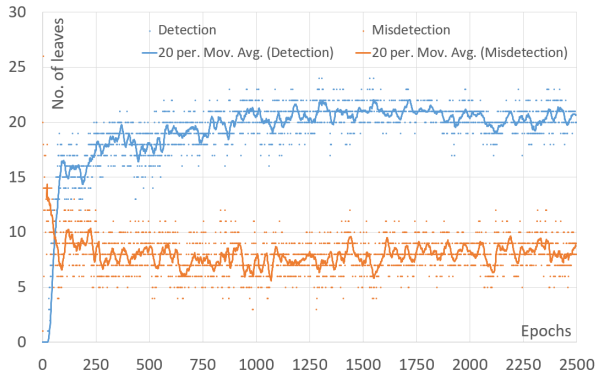

Figure 11: Detection vs misdetection rates on avocado leaves, by training epochs.

Performance in the Leaf Segmentation and Counting Challenges is evaluated by several criteria, fully described in [6]. Since our main goal is accurate leaf contouring, decision was to focus the BestDice score, measuring degree of overlap between leaf segmentation results and the ground truth. First we validate the network's performance with the evaluation script supplied by the challenge, training each subset separately. This allows us to choose the best epoch for each subset and run the test with this epoch's weights. Visualization of a training example from the arabidopsis A1 category, its ground-truth leaf masks and the network's segmentation results are presented in Figure 12.

Table 2: Evaluation scores

	A1	A2	A3	A4	A5	Mean
BestDice	88.7	84.8	83.3	88.6	85.9	86.7
FgBgDice	89.1	87.9	82.3	88.4	87.0	87.1
AbsDiffFG	5.30	1.89	2.01	4.81	4.01	4.11
DiffFG	-5.30	-1.67	-1.69	-4.81	-3.88	-4.01

The network's performance, by training epochs, is presented in Figure 13 (BestDice evaluation) and Figure 14

Figure 12: Structured collage result example: Left – training image, Center – training mask, Right – network segmentation of the image.

Table 3: Segmentation performance comparison (BestDice)

	A1	A2	A3	A4	A5	Mean
Romera [36]	66.6	-	-	-	-	-
Pape [29]	74.4	76.9	53.3	-	-	62.6
Pape [30]	80.9	78.6	64.5	-	-	71.3
Salvador [37]	74.7	-	-	-	-	-
De Brabandere [7]	84.2	-	-	-	-	-
Ren [33]	84.9	-	-	-	-	-
Zhu [50]	-	-	-	87.9	-	-
Ward [48]	**90**	81	59	88	82	81
Ours	88.7	**84.8**	**83.3**	**88.6**	**85.9**	**86.7**

(absolute difference in count). Although we expect to see some over-fitting in the evaluation graph (recalling that training and validation data are different), the network's performance remains stable from epoch 400, as can be seen in both figures. Evidently, scores on the A2 and A3 subsets are poorer in segmentation accuracy and better at counting, compared to A1 and A4. A possible explanation is the smaller sets of extracted leaves from the A2,A3 categories that the network was trained on. Since the optimization loss is a combination of five functions, this scenario leads the network learning, while struggling to improve segmentation loss, to direct its efforts on count loss. Table 2 presents our results on all the A1-A5 datasets and mean over all subsets. Table 3 compares BestDice evaluation of segmentation performance. Note that most works report results on the A1 subset only, and that the A4,A5 subsets were added at a later date. Few other features were tried for improving the obtained results, most notable alpha blending of leaf boundaries and blurring the boundaries using mean or Gaussian filter. Although some of these attempts enhance image realism and especially boundaries between the objects, they did not lead to improved results. In spite of this fact, our recommendation is to continue research in this direction since the weakest point of the current state is separation between two (or more) overlapping leaves. Specifically, we suspect that fine tuning of leaf boundary adjustment will lead to closer correlation between the synthesized and real images.

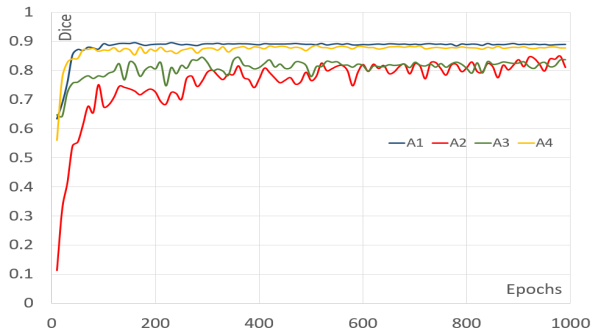

Figure 13: BestDice evaluation on training data on the four categories of the dataset.

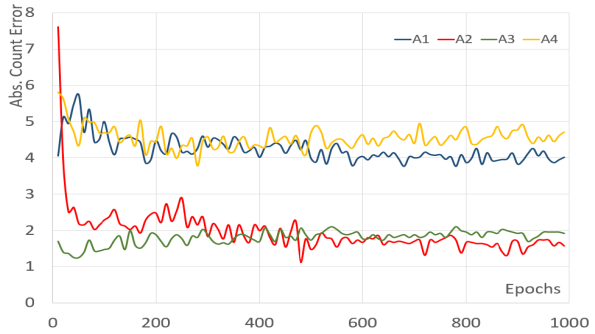

Figure 14: Absolute difference of counting error on the four categories of the dataset.

5. Conclusions

We have shown that data augmentation preserving geometric features and sophisticated positioning of objects enhances network performance in the tasks of object instance detection and segmentation. The suggested method was tested on the publicly available dataset of rosette plants, and achieved high scores on the leaf segmentation and counting tasks. We hope this modest contribution will serve to motivate further investigation of integrating synthetic data augmentation with real world botanical scenes for various plant phenotyping tasks. Similar structured collaging techniques may well be adapted to other domains, such as autonomous navigation, urban modeling, satellite and medical imagery.

Acknowledgments

The authors thank Ortal Bakhshian for collecting and annotating the avocado images. This research was partly supported by the Israel Innovation Authority, the Phenomics Consortium.

References

[1] S. Aich and I. Stavness. Leaf counting with deep convolutional and deconvolutional networks. In *Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy*, pages 22–29, 2017. 3

[2] N. M. Al-Vshakarji, Y. M. Kassim, and K. Palaniappan. Unsupervised learning method for plant and leaf segmentation. In *2017 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–4. IEEE, 2017. 3

[3] S. Arvidsson, P. Pérez-Rodríguez, and B. Mueller-Roeber. A growth phenotyping pipeline for arabidopsis thaliana integrating image analysis and rosette area modeling for robust quantification of genotype effects. *New Phytologist*, 191(3):895–907, 2011. 3

[4] J. Bell and H. Dee. Aberystwyth leaf evaluation dataset. *URL: https://doi. org/10.5281/zenodo*, 168158:17–36, 2016. 2

[5] A. Camargo, D. Papadopoulou, Z. Spyropoulou, K. Vlachonasios, J. H. Doonan, and A. P. Gay. Objective definition of rosette shape variation using a combined computer vision and data mining approach. *PLoS One*, 9(5):e96889, 2014. 3

[6] Cvppp data description. https://www.plant-phenotyping.org/lw_resource/datapool/systemfiles/elements/files/b05bc767-348a-11e7-8c78-dead53a91d31/live/document/LSC_2017_data_description_and_further_details.pdf. 7

[7] B. De Brabandere, D. Neven, and L. Van Gool. Semantic instance segmentation with a discriminative loss function. *arXiv preprint arXiv:1708.02551*, 2017. 3, 8

[8] A. Dobrescu, M. V. Giuffrida, and S. A. Tsaftaris. Leveraging multiple datasets for deep leaf counting. In *Computer Vision Workshop (ICCVW), 2017 IEEE International Conference on*, pages 2072–2079. IEEE, 2017. 3

[9] C. Douarre, R. Schielein, C. Frindel, S. Gerth, and D. Rousseau. Transfer learning from synthetic data applied to soil–root segmentation in x-ray tomography images. *Journal of Imaging*, 4(5):65, 2018. 1

[10] N. Dvornik, J. Mairal, and C. Schmid. Modeling visual context is key to augmenting object detection datasets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 364–380, 2018. 1

[11] N. Dvornik, J. Mairal, and C. Schmid. On the importance of visual context for data augmentation in scene understanding. *arXiv preprint arXiv:1809.02492*, 2018. 1

[12] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361. IEEE, 2012. 1

[13] Gimp. https://github.com/bootchk/resynthesizer. 5

[14] M. V. Giuffrida, P. Doerner, and S. A. Tsaftaris. Pheno-deep counter: a unified and versatile deep learning architecture for leaf counting. *The Plant Journal*, 96(4):880–890, 2018. 3

[15] M. V. Giuffrida, M. Minervini, and S. Tsaftaris. Learning to count leaves in rosette plants. In H. S. S. A. Tsaftaris and T. Pridmore, editors, *Proceedings of the Computer Vision Problems in Plant Phenotyping (CVPPP)*, pages 1.1–1.13. BMVA Press, September 2015. 3

[16] M. V. Giuffrida, H. Scharr, and S. A. Tsaftaris. Arigan: Synthetic arabidopsis plants using generative adversarial network. In *Proceedings of the 2017 IEEE International Conference on Computer Vision Workshop (ICCVW), Venice, Italy*, pages 22–29, 2017. 3

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 3

[18] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1–8. IEEE, 2009. 1

[19] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017. 1, 7

[20] Y. Itzhaky, G. Farjon, F. Khoroshevsky, A. Shpigler, and A. B. Hillel. Leaf counting: Multiple scale regression and detection using deep cnns, 2018. 3

[21] A. Kamilaris and F. X. Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147:70–90, 2018. 1

[22] E. Kaminuma, N. Heida, Y. Tsumoto, N. Yamamoto, N. Goto, N. Okamoto, A. Konagaya, M. Matsui, and T. Toyoda. Automatic quantification of morphological traits via three-dimensional measurement of arabidopsis. *The Plant Journal*, 38(2):358–365, 2004. 3

[23] A. Lindenmayer. Mathematical models for cellular interactions in development i. filaments with one-sided inputs. *Journal of theoretical biology*, 18(3):280–299, 1968. 2

[24] Matterport. https://github.com/matterport/Mask_RCNN. 6

[25] M. Minervini, A. Fischbach, H. Scharr, and S. Tsaftaris. Plant phenotyping datasets, 2015. 2, 6

[26] M. Minervini, A. Fischbach, H. Scharr, and S. A. Tsaftaris. Finely-grained annotated datasets for image-based plant phenotyping. *Pattern recognition letters*, 81:80–89, 2016. 2

[27] L. Mündermann, Y. Erasmus, B. Lane, E. Coen, and P. Prusinkiewicz. Quantitative modeling of arabidopsis development. *Plant physiology*, 139(2):960–968, 2005. 2

[28] S. T. Namin, M. Esmaeilzadeh, M. Najafi, T. B. Brown, and J. O. Borevitz. Deep phenotyping: deep learning for temporal phenotype/genotype classification. *Plant methods*, 14(1):66, 2018. 3

[29] J.-M. Pape and C. Klukas. 3-d histogram-based segmentation and leaf detection for rosette plants. In *European Conference on Computer Vision*, pages 61–74. Springer, 2014. 3, 8

[30] J.-M. Pape and C. Klukas. Utilizing machine learning approaches to improve the prediction of leaf counts and individual leaf segmentation of rosette plant images. *Proceedings of the Computer Vision Problems in Plant Phenotyping (CVPPP)*, pages 1–12, 2015. 3, 8

[31] P. Prusinkiewicz and A. Lindenmayer. *The algorithmic beauty of plants*. Springer Science & Business Media, 2012. 2

[32] Rahan. http://www.rahan.co.il/. 2

[33] M. Ren and R. S. Zemel. End-to-end instance segmentation with recurrent attention. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA*, pages 21–26, 2017. 3, 8

[34] E. Richardson, M. Sela, and R. Kimmel. 3d face reconstruction by learning from synthetic data. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 460–469. IEEE, 2016. 1

[35] E. Richardson, M. Sela, R. Or-El, and R. Kimmel. Learning detailed face reconstruction from a single image. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 5553–5562. IEEE, 2017. 1

[36] B. Romera-Paredes and P. H. S. Torr. Recurrent instance segmentation. In *European Conference on Computer Vision*, pages 312–329. Springer, 2016. 3, 8

[37] A. Salvador, M. Bellver, V. Campos, M. Baradad, F. Marques, J. Torres, and X. Giro-i Nieto. Recurrent neural networks for semantic instance segmentation. *arXiv preprint arXiv:1712.00617*, 2017. 3, 8

[38] H. Scharr, M. Minervini, A. Fischbach, and S. A. Tsaftaris. Annotated image datasets of rosette plants. In *European Conference on Computer Vision. Zürich, Suisse*, pages 6–12, 2014. 2

[39] H. Scharr, M. Minervini, A. P. French, C. Klukas, D. M. Kramer, X. Liu, I. Luengo, J.-M. Pape, G. Polder, D. Vukadinovic, et al. Leaf segmentation in plant phenotyping: a collation study. *Machine vision and applications*, 27(4):585–606, 2016. 2

[40] M. Sela, E. Richardson, and R. Kimmel. Unrestricted facial geometry reconstruction using image-to-image translation. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 1585–1594. IEEE, 2017. 1

[41] A. K. Singh, B. Ganapathysubramanian, S. Sarkar, and A. Singh. Deep learning for plant stress phenotyping: trends and future perspectives. *Trends in plant science*, 2018. 1

[42] R. Slossberg, G. Shamai, and R. Kimmel. High quality facial surface and texture synthesis via generative adversarial networks. *arXiv preprint arXiv:1808.08281*, 2018. 1

[43] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 23–30. IEEE, 2017. 2

[44] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. *arXiv preprint arXiv:1804.06516*, 2018. 2

[45] S. A. Tsaftaris and H. Scharr. Sharing the right data right: A symbiosis with machine learning. *Trends in plant science*, 2018. 2

[46] J. Ubbens, M. Cieslak, P. Prusinkiewicz, and I. Stavness. The use of plant models in deep learning: an application to leaf counting in rosette plants. *Plant methods*, 14(1):6, 2018. 2, 3, 6

[47] J. R. Ubbens and I. Stavness. Deep plant phenomics: a deep learning platform for complex plant phenotyping tasks. *Frontiers in plant science*, 8:1190, 2017. 2

[48] D. Ward, P. Moghadam, and N. Hudson. Deep leaf segmentation using synthetic data. *arXiv preprint arXiv:1807.10931*, 2018. 2, 3, 8

[49] X. Yin, X. Liu, J. Chen, and D. M. Kramer. Joint multi-leaf segmentation, alignment, and tracking for fluorescence plant videos. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1411–1423, 2018. 3

[50] Y. Zhu, M. Aoun, M. Krijn, J. Vanschoren, and H. T. Campus. Data augmentation using conditional generative adversarial networks for leaf counting in arabidopsis plants. *Computer Vision Problems in Plant Phenotyping (CVPPP2018)*, 2018. 3, 8