Exploration of Data Augmentation Techniques for Bush Detection in Blueberry Orchards

Boris Čuljak, Nina Pajević, Vladan Filipović, Dimitrije Stefanović,
Željana Grbović, Nemanja Djuric, Marko Panić
BioSense Institute, Novi Sad, Serbia
culjak.ra31.2020@uns.ac.rs, {nina.pajevic, vladan.filipovic, dimitrije.stefanovic, zeljana.grbovic}@biosense.rs, nemanja@temple.edu, panic@biosense.rs

Abstract

Advancements in object detection technology have led to its widespread application across various fields, yet its adoption in agriculture, particularly for precision tasks like orchard navigation and crop monitoring, has not been fully realized. Our research extends the dialogue on agricultural applications by focusing on the vital role of data augmentation techniques in enhancing the detection of blueberry bushes, a critical part of smart farming in blueberry orchards. Utilizing a data set that captures blueberry bushes under diverse environmental conditions, we conduct an in-depth analysis of how different data augmentation strategies affect the performance and robustness of bush detection models. We present a comparative study to understand the impact of such techniques, and propose a combined data augmentation that outperforms individual approaches. Our findings establish benchmarks for model performance on this task, and also illuminate the path forward for improving advanced detection methods in general agricultural applications. By detailing the efficacy of various augmentation methods, we aim to spur further innovation in agricultural technology, thus helping the community move towards more efficient and intelligent farming practices.

1. Introduction

Agriculture stands as a cornerstone of global food security, economic development, and sustainability. It is a sector that directly influences the lives of billions of people, focusing on ensuring a stable supply of food, generating employment, and fostering socio-economic growth [2, 7]. Despite its critical importance, agriculture faces numerous challenges, including climate change, resource scarcity, and increasing demands from a growing global population [1, 26, 37]. The integration of advanced technologies, including object detection and other machine learning methods, offers promising solutions to revolutionize traditional farming practices, leading to improved crop management, yield prediction, and resource allocation [21, 29, 40]. Moreover, generative models have recently emerged as a powerful tool in this technological arsenal, enabling the creation of synthetic agricultural data that can further enhance the training of machine learning models, thereby improving their ability to generalize across diverse and complex agricultural scenarios [25]. As such, further advancing object detection applications and exploring the role of generative models within agriculture is essential for harnessing the potential of technology to help secure future food supplies and promote environmental stewardship.

Data augmentation represents a critical methodology in the development of robust machine learning models, particularly in the domain of object detection. In agriculture, the variability of environmental conditions, plant appearances, and growth stages poses significant challenges for automated systems. Data augmentation techniques, which artificially expand the diversity of training data through modifications and transformations, are crucial for improving model generalization and performance. In particular, by simulating a wider range of conditions and scenarios, these techniques enable detection models to become more resilient to real-world variability, enhancing their accuracy and reliability in agricultural applications [27]. This is paramount for tasks such as crop disease detection [14], precision weed control [15], or fruit counting [35], to name a few, where the precision of detection models directly impacts decision-making processes and operational efficiency. The importance of data augmentation in agriculture extends beyond model performance, facilitating the adoption of autonomous systems and smart farming solutions capable of adapting to complex agricultural environments. In this context, our study explores the effects of various data augmentation techniques on the performance of detection models, with a special focus on blueberry bush detection.

Furthermore, robots and automation have emerged as indispensable tools for ensuring productivity and success in agriculture today, particularly in large fields and amidst escalating labor shortages. Unmanned ground vehicles (UGVs) are increasingly prominent in agricultural robotics due to their ability to integrate sensors and imaging technology, facilitating crop assessment and assisting in activities like fertilization, seeding, and weed management [11]. Accurate bush detection in images captured with a camera mounted on a UGV is a requirement for the task of autonomous vehicle guidance in orchards, where GPS information may not be reliable or precise enough [9], as well as for other tasks such as precise weed control [10]. Recognizing the value of blueberry crops and the challenges associated with their cultivation, our research aims to advance the application of autonomous robots and precision agriculture. As such, through a rigorous examination of data augmentation, we seek to not only enhance model efficacy in detecting blueberry bushes but also contribute to the broader field of agricultural innovation, where the potential for technological intervention remains largely untapped [22].

We summarize the contributions of our work below:

- we provide a study of how different data augmentation techniques impact the performance of bush detectors;
- following this exploration, we propose a data augmentation combination that outperforms the individual augmentation approaches.

2. Related Work

In this section we give an overview of the landscape of data augmentation techniques within the field of object detection, with a particular focus on agricultural applications.

2.1. Data Augmentation

The importance of extensive data sets for training deep learning models is paramount [13], a principle that holds especially true in areas that involve complex interactions such as those encountered in agriculture [19]. Data augmentation is a critical technique in this context, serving to artificially enhance the size and diversity of training data sets. This is commonly achieved through the application of various transformations to existing data, thereby generating new data points. Such methods are crucial for improving the performance of deep learning models, as they help to mitigate over-fitting, bolster the models' ability to generalize, and address challenges associated with limited data availability [31]. By introducing a broader array of variations to the training data, models are trained to better handle the intricacies and unpredictability of real-world applications, enhancing their accuracy and reliability across different scenarios [20]. In this work we explore the use of data augmentation for the task of object detection, as applied to the important agricultural task of bush detection.

2.2. Agricultural Applications

In the realm of agriculture, high variability of environmental conditions and the scarcity of data compound the challenges of applying deep learning models effectively. Traditional data augmentation techniques, such as color and geometric transformations, emerged as vital tools in this domain [4, 30]. These methods adapt models to the diverse lighting conditions and perspectives under which agricultural data is captured, effectively expanding the range of conditions represented in the training data sets and resulting in improved model performance [28]. Specifically, in the context of UGV-captured data, data augmentation becomes crucial due to the inherently slow and costly nature of data collection, coupled with the low variability in the obtained data. This practice is instrumental in enhancing the robustness and accuracy of object detection models used for essential tasks like crop monitoring and disease detection [24]. Moreover, innovative approaches to data augmentation, such as the creation of synthetic images through techniques like collage [34] or mosaic generation [5], have shown promise in further enhancing the performance of convolutional object detectors within the complex and non-structured environments typical for agriculture. These methods were shown to preserve the texture and context of target objects against realistic backgrounds, resulting in substantial improvement in model precision. This earlier work underlines the critical role of diverse and extensive data sets in developing robust object detection models for agricultural applications.

2.3. Generative Augmentation

As the agricultural sector increasingly adopts advanced technologies to surmount its myriad challenges, there has been a notable uptick in research exploring the integration of generative networks [8, 16, 41]. The diversity of architectures within the realm of Generative Adversarial Networks (GANs) [25], for instance, highlights the versatility and potential of such approaches. These developments underscore the potential of generative networks to enrich training data with synthetic, highly realistic agricultural data, offering a previously unattainable level of data diversity [3].

On the other hand, traditional augmentation methods such as geometric transformations have been shown to reach a comparable or better performance than generative models on certain tasks [31]. This is understood to largely be due to challenges in generating agricultural images that are high in both quality and realism. Nevertheless, research and development in this field positions generative augmentation as a pivotal future direction for agricultural applications. The potential of these models to generate an unlimited array of realistic images could profoundly transform the landscape of data availability, greatly improving the training, robustness, and efficacy of deep models in agriculture, which is a direction we also explore in our current work.

(a) Color image augmentations: hue adjustment; saturation adjustment; value adjustment; combined adjustments

(b) Geometric image augmentations: translated image; scaled image; flipped image; combined adjustments

(c) Mosaic augmentation (bounding boxes indicate bush labels)

Figure 1. Examples of color, geometric, and mosaic image augmentation

3. Data Augmentation for Bush Detection

In this section we explore a broad spectrum of data augmentation techniques applicable to the task of bush detection in blueberry orchards, ranging from standard image transformations to sophisticated generative-based ones.

3.1. Color Augmentations

Color augmentations are a set of techniques used in image processing to modify the colors in images to increase the diversity of the training data and improve model generalization. These techniques, which include adjustments in brightness, saturation, and hue, effectively simulate a variety of lighting conditions and color settings that might not have been present in the original data (illustrated in Fig. 1a).

Given the practical difficulties of capturing images at every time of day and season, in the context of bush detection color augmentations can be particularly useful. They allow for the simulation of different times of the year and day, such as creating a more grayish tone to mimic winter, enhancing brightness for a summery feel, or adjusting the light to recreate morning or evening scenes. This versatility enhances the model's ability to interpret images accurately across diverse conditions, making it more robust to changes in lighting conditions and color distributions.

3.2. Geometric Augmentations

In enhancing our model using geometric augmentations, we focused on techniques expected to impact agricultural applications, and especially bush detection (see Fig. 1b). In particular, translation augmentation is tailored to mimic the shifting perspectives an UGV might encounter due to uneven terrain or between the dense foliage of blueberry bushes, ensuring reliable detection even when interacting with plants, fruits, or other obstacles. This prepares the UGV to accurately identify targets, regardless of their position in the vehicle's field of view, enhancing operational efficiency and precision. Scale and flip augmentations adjust the size and orientation of objects in training images, vital for recognizing the varied sizes and shapes of blueberry bushes. This helps the UGV accurately assess and interact with different parts of the orchard, from individual berries to entire bushes, and handle tasks regardless of the object's size or orientation.

3.3. Mosaic Augmentations

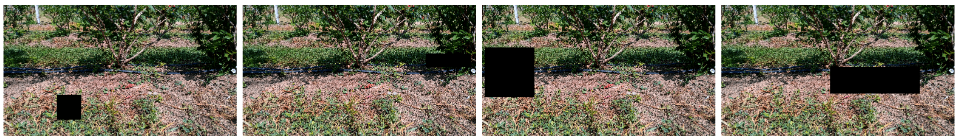
Mosaic image augmentation is a technique designed to stitch together multiple images into a single composite training sample. This approach is very useful in enhancing detection models by introducing a diverse array of visual contexts and scenarios within a single image frame (see Figure 1c, where zeros represent the bush class). It allows models to encounter and learn from a variety of object positions, scales, and environmental conditions, thereby improving their ability to generalize from the training data to real-world situations.

In applications like ours, mosaic augmentation can be quite useful. It exposes the model to complex environments where blueberry bushes might appear in different configurations and densities. However, the process of stitching multiple images together in mosaic augmentation can sometimes result in unintended and unrealistic scenarios. For example, blueberry bushes from one segment might appear as if they are growing on top of the sky in another segment, creating visually incongruent scenarios.

3.4. Gaussian Noise Augmentation

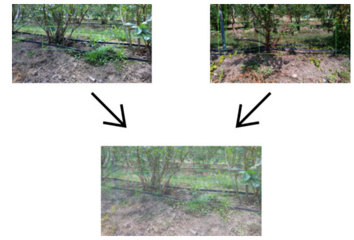
Gaussian noise augmentation involves superimposing a layer of Gaussian noise over original images (see Fig. 2a). This noise, characterized by its bell-shaped curve centered around a zero mean, introduces random variations to the pixels, emulating the type of visual disturbances that might occur in real-world imaging scenarios due to issues such as sensor fouling or other hardware deficiencies.

(a) Gaussian noise image augmentations: σ=50, Frequency=1; σ=100, Frequency=1; σ=50, Frequency=0.5; σ=100, Frequency=0.5



(b) Cutout image augmentations: Area=0.02, Aspect=1:1; Area=0.02, Aspect=1:3; Area=0.08, Aspect=1:1; Area=0.08, Aspect=1:3

(c) Collage image augmentations



(d) Mix-up augmentation $\lambda = 0.5$

(e) DCGAN generated images of bushes

Figure 2. Examples of Gaussian noise, cutout, collage, mix-up, and DCGAN image augmentation

The effect of this noise can be controlled using frequency and σ parameters. In particular, σ determines the standard deviation of the noise values, affecting the intensity and spread of Gaussian noise added to the pixels. On the other hand, frequency controls how often the noise is applied, with a value of 1 indicating it is applied to every pixel, and values closer to 0 resulting in sparser noise distribution across the image. For UGVs operating in orchards, introducing Gaussian noise during training helps prepare the model for the inevitable variations in lighting, weather conditions, and sensor imperfections they could encounter.

3.5. Cutout Augmentation

Cutout augmentation is a technique that enhances the ability of image-processing models to handle occlusions and partial visibility by randomly erasing sections of an image. This method effectively simulates real-world disruptions, such as foliage, branches, or equipment partially covering the target objects, by replacing selected image regions with zeroes and thus creating artificial occlusions (see Fig. 2b). As seen in the figure, method parameters include area and aspect, with area dictating the percentage of the image area to be cut out, and aspect determining the aspect ratio of the cutout box. This approach helps the model robustly and reliably detect and classify blueberry bushes even when they are not fully visible, a common challenge in dynamic and complex orchard environments.

3.6. Collage Augmentation

This method involves placing objects, such as cutouts of blueberry bushes, onto entirely different scenes. These cutouts, accurately annotated and extracted from their original context, are strategically positioned in the lower third of the new background, to mirror the real-world positioning of ground-truth labels (see Fig. 2c). This augmentation strategy creates diverse and complex scenes by introducing the objects of interest into a variety of backgrounds, thereby challenging the model to recognize and classify these objects in contexts outside of their common setting. The randomness in the number of objects added and their placement ensures that the model is exposed to a wide range of scenarios, enhancing its ability to identify blueberry bushes across different backgrounds and under various conditions.

3.7. Mix-up Augmentation

Mix-up approach [38] was created for classification tasks, and it has since been shown to work well in object detection applications as well [44]. This approach essentially blends two images and their labels together based on a weighting parameter λ (see Fig. 2d). By blending the data from two separate training examples, mixup encourages the model to learn more robust and generalizable features that are applicable across different variations of blueberry bushes, thereby improving its ability to detect them accurately in diverse and noisy environments.

3.8. DCGAN Augmentation

When it comes to augmentation using generative models, in this work we focused on applying Deep Convolutional GAN (DCGAN) for the specific purpose of generating bush labels, training the method on the bush examples from the non-augmented data. Generating images with a DCGAN involves an interplay between two neural networks: the generator and the discriminator, which are trained adversarially to each other. The generator's role is to create images indistinguishable from the real ones, while the discriminator's task is to distinguish between the generated (i.e., fake) images and actual (i.e., real) images. Over time, this competition drives both networks to improve their capabilities, with the generator producing increasingly more realistic images throughout the training [12].

The key aspect of DCGAN lies in its use of deep convolutional networks, which are particularly adept at handling image data. Convolutional layers are designed to recognize and extract patterns and features from images, such as edges, textures, and more complex shapes, making them highly effective for image-related tasks. In the context of agriculture, DCGAN is expected to be useful for generating high-quality, diverse, and realistic images of crops, plants, or agricultural scenarios (see Figure 2e for examples).

4. Experiments

In this section, we focus on empirical evaluation that sheds light on the role of data augmentation in agriculture. We outline the experimental setup, including the models used, data analysis, and the specific augmentation techniques applied. Our discussion will cover the obtained results and what these outcomes mean for agricultural applications.

4.1. Data

The data set used is the blueberry orchard data [9] totaling 7,250 training, test, and validation images. The data contains images of a blueberry orchard, captured with an RGB camera with a resolution of 1920×1080 pixels mounted on a UGV. The images represent blueberry bushes as seen with a back-view camera pointed down the row and a side-view camera pointed directly at the bushes, and were captured during the UGV's ride between different rows. The images were created during two growing seasons, at different phenological stages, and captured at different times of day and illumination conditions.

The objects of interest in the images are blueberry bushes, where labels frame the base of the bush in contact with the ground, as well as poles (such as hail netting poles) where labels include as much of the pole as possible, including the contact of the pole and the ground. This collection encompasses a total of 24,830 bush labels and 2,986 pole labels, with the height and width distribution of the labels illustrated in Figure 3. Lastly, the data set is divided into different splits: approximately 75% (5,437 images) for training, 10% (725 images) for validation, and around 15% (1,088 images) for testing.

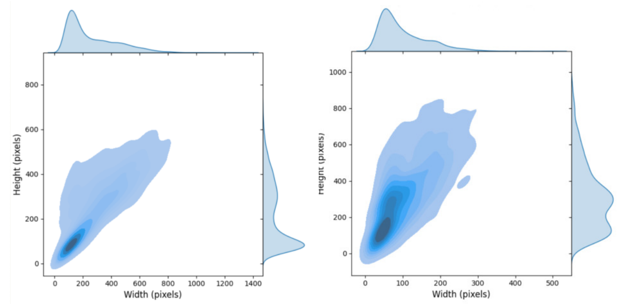


Figure 3. Width and height distribution of bush and pole labels

4.2. Augmentation Approach

We aimed to increase the overall data size by 30% to ensure a uniform augmentation strategy across the approaches. In most cases this involved randomly selecting a required number of original images, running the augmentation method, and reintegrating the resulting images into training and validation sets to be used during training. In the remainder of this section, we provide details on each augmentation approach used to enrich the original data:

- Color Augmentations: We adjusted the color properties of each selected image by changing the HSV (Hue, Saturation, Value) values according to a random percentage of their original values, within predefined ranges of ±1.5%, ±70%, and ±40%, respectively.
- Geometric Augmentations: Our approach included geometric transformations such as translation, scaling, and horizontal flipping. All selected images were randomly shifted within a ±10% range of their dimensions and resized within a ±50% range of their original scale. Additionally, images were flipped horizontally with a 50% probability, mimicking various orientations and perspectives that objects can assume in real-world conditions.
- Mosaic Augmentation: As a part of this technique we randomly selected four images to generate one additional image, repeating this process multiple times to achieve the required data increase.
- Gaussian Noise Augmentation: To replicate the unpredictable nature of environmental noise, we introduced Gaussian noise with a zero mean and a standard deviation ranging between 30 and 70. This augmentation was applied to images where pixel values ranged from 0 to 255, at a frequency of 0.8.
- Cutout Augmentation: This technique randomly erased a single part of the selected image, which was then filled

Table 1. Experimental results for the considered augmentation techniques; bolded numbers denote best performance per metric; dashed line separates single-augmentation approaches from approaches using multiple augmentations

Method	All					Bush					Pole				
	P	R	F1	mAP 50	mAP 50-95	P	R	F1	mAP 50	mAP 50-95	P	R	F1	mAP 50	mAP 50-95
None	0.750	0.650	0.720	0.480	0.696	0.800	0.680	0.740	0.500	0.735	0.700	0.620	0.680	0.460	0.658
Color	0.790	0.690	0.750	0.510	0.737	0.830	0.710	0.780	0.530	0.765	0.750	0.670	0.710	0.490	0.708
Geometric	0.810	0.710	**0.780**	**0.550**	0.757	0.840	0.730	0.800	**0.570**	**0.781**	0.780	0.690	0.740	**0.530**	0.732
Mosaic	0.800	**0.720**	0.760	0.530	**0.758**	0.820	**0.740**	0.790	0.550	0.778	0.780	**0.700**	0.740	0.520	0.738
Gaussian	0.770	0.670	0.730	0.490	0.717	0.790	0.690	0.750	0.510	0.737	0.780	0.700	0.740	0.520	0.738
Cutout	0.760	0.660	0.710	0.470	0.706	0.780	0.680	0.730	0.490	0.727	0.740	0.640	0.690	0.460	0.686
Collage	0.729	0.671	0.728	0.471	0.699	0.777	0.680	0.761	0.461	0.725	**0.826**	0.689	**0.795**	0.451	**0.751**
Mix-up	**0.832**	0.601	0.742	0.473	0.698	**0.910**	0.634	**0.805**	0.478	0.747	0.753	0.567	0.679	0.468	0.647
DCGAN	0.755	0.642	0.706	0.420	0.694	0.854	0.668	0.803	0.463	0.750	0.655	0.616	0.609	0.378	0.635
Uniform	0.829	0.656	0.732	0.766	0.461	0.831	0.700	0.760	0.815	0.461	**0.827**	0.612	0.703	0.716	0.461
Combo	0.775	0.712	0.742	**0.792**	0.475	0.844	0.755	0.797	**0.849**	0.470	0.706	0.670	0.688	**0.735**	0.480
Full-combo	**0.853**	**0.803**	**0.884**	0.615	**0.827**	**0.893**	**0.767**	**0.877**	0.581	**0.825**	0.813	**0.839**	**0.890**	0.648	**0.826**

with zeros. The size of the erased section varied randomly from 1% to 4% of the total image area, with random aspect ratios ranging from 0.8 to 3, simulating realistic occlusions found in natural environments.

- Collage Augmentation: We generated new scenes by overlaying bush and pole cutouts, extracted from various images, onto random background images without preexisting objects. This process varied the scene complexity by altering the number of superimposed objects, ranging randomly from 1 to 5, positioned in the lower third of the image to mirror the natural placement of the objects.
- Mix-up Augmentation: This technique blended two randomly chosen images into one by averaging them together using $\lambda = 0.5$.
- DCGAN Augmentation: We focused on generating individual bush labels, aiming to increase the total number of bush labels by 30% and upweighting pole and negative labels by the same amount. Synthetic bush images were created at a 128x128 resolution. However, to closely replicate the dimensions of the original labels, we further resized the generated images with a random deviation between 0 to 30 pixels. Further details about DCGAN training and integration can be found in the appendix.

4.3. Detector Model

We employed the YOLOv8 ”nano” version (YOLOv8n) as our object detector, shown to be an efficient and effective solution in real-time applications [6, 42, 43]. The YOLOv8n model, a progression from the earlier YOLOv5 [39] architecture which has long been a staple for similar object detection tasks [9], is distinguished by its lean design, incorporating 3.2 million parameters that facilitate a good balance between compactness and performance [18]. This model mirrors the detection accuracy of its YOLOv5n predecessor, yet stands out significantly for its training speed, achieving the same results in about one-third of the time [17]. This high efficiency of the detector is critical to ensure real-time guidance and decision-making of the UGV.

The training process was initiated from a model pretrained on the COCO data, and was run for 50 epochs. We maintained the default input resolution at 640×640 pixels, resizing the images to match this specification. During training we minimized a composite loss function, integrating three pivotal components: classification loss, object loss, and location loss [39, 45]. Both classification and objectness losses are computed using binary cross-entropy, while the location loss is derived from the complete intersection-over-union (IoU) loss [39, 45]. This multifaceted loss function plays a critical role in refining the model’s ability to accurately identify and localize objects within an image. Lastly, we used a batch size of 32 and a learning rate of 0.01, employing the Adam optimizer [23].

For the evaluation step, the detection confidence threshold was set to 0.24, while the value of the matching IOU threshold was set to the default value of 0.6. Furthermore, Non-Maximum Suppression (NMS) with an IOU threshold of 0.6 was applied to the model output to eliminate duplicate detections [36].

4.4. Results

We used precision, recall, F1 score, mAP50, and mAP50-95 to evaluate the approaches [32, 45], with the results shown in Table 1. Here, mAP50 assesses the model performance at a 50% IoU threshold, while mAP50-95 averages performance across a range of IoU thresholds from 50% to 95%.

Compared to the approach without any augmentation, we can see that the introduction of color augmentations offered only a slight increase in metrics, This can be explained by the uniformity of the test images, which were all captured under similar sunny conditions. This homogeneity meant there was limited scope for improvement through variations in image brightness and saturation. However, we can see that a boost in performance was observed with geometric augmentations such as translation, flipping, and scaling, markedly improving precision in object localization, evidenced by increases in both mAP50 and mAP50-95 metrics. This notable enhancement can be attributed to the nature of the data collection process. The UGV's mobility inherently introduces variability in the perspective, positioning, and orientation of the captured objects, with the detector thus benefiting from this augmentation technique.

Mosaic augmentation brought about a notable enhancement in recall rates across all categories, with an average increase of about 10%. This technique helped the model generalize better by creating randomized composite scenes, making it more adept at identifying objects under varied conditions. Moreover, Gaussian augmentation managed to match or slightly exceed the baseline metrics, though not yielding substantial improvements.

We can also see that collage augmentation presented a nuanced outcome. While generally aligning with baseline performances, it notably boosted the precision, F1 score, and mAP50-95 for the pole category. This suggests that the original dataset may have had a limited representation of poles compared to bushes, highlighting a potential imbalance or deficiency in the data set. Furthermore, mix-up encourages the model to focus on more generalized features rather than memorizing specific details of the training images. By learning from these blended images, the model becomes better at identifying the presence of objects within an image, even when they are partially obscured or presented in a less typical context. Given this approach, it is not surprising that the precision metrics across all categories experienced a boost of more than 10%.

Lastly, the experimentation with DCGAN-generated images produced mixed results, with performance occasionally outperforming the baseline approach. The variability points to the nature of the generated images lacking realism, which might have contributed to the fluctuating outcomes. Further improving the realism of these synthetic images could potentially enhance their effectiveness.

4.4.1 Proposed Augmentation Strategies

After careful analysis of the unique benefits provided by each augmentation technique, we decided to select and evaluate a combination of geometric, mosaic, collage, and mix-up augmentations as a new strategy. This was motivated by the analysis detailed in Table 1, where the best result for each metric is highlighted in bold, underscoring the augmentation that contributed most significantly to it.

We implemented the above-mentioned augmentations in three distinct variants, listed below the dashed line in Table 1 as "Uniform", "Combo", and "Full-combo" approaches. The uniform approach involved applying each of the four chosen augmentation techniques uniformly to one-quarter of the additional 30% of data across our train and validation sets. In the combo approach, all four augmentations were applied concurrently to each image in the additional 30% of data. In this approach each image of the expansion was augmented first with the collage augmentation technique, followed by the geometric, mix-up, and finally the mosaic technique. Lastly, for the full-combo method the 30% increase was generated exclusively using the collage augmentation, after which the remaining three augmentations were applied to every image in the entire data set (including both the original and the expanded data), in the same order as in the combo approach.

We can see that the combo approach generally outperformed the uniform method in most metrics, suggesting better generalization capabilities due to the more diverse training data that it provides. However, it struggles with precision compared to the uniform method, potentially indicating that the extensive modifications from multiple augmentations might distort the original images too significantly. Motivated by the combo method's superior performance over the uniform strategy in most metrics, we decided to investigate the full-combo approach. This method not only improved overall generalization but also enhanced precision, which is a notable contrast to the combo method's struggle in this area. We theorize that the improved precision in the full-combo method could be attributed to the larger percentage of the overall training data consisting of augmented images, thus providing more comprehensive learning examples. Despite these advancements, we notice that the full-combo method exhibited a lower mAP50 compared to both the uniform and combo methods, indicating a potential trade-off between extensive augmentation and optimal detection thresholds across different object scales.

To conclude, we can see that the full-combo strategy resulted in significant enhancements across the board for each object category. This result not only underscores the strength of the proposed augmentation approach, but also emphasizes the importance of employing an augmentation strategy that is tailored specifically for the problem domain in question (as we rely on a combination of augmentations shown to provide benefits for the agricultural task at hand).

4.4.2 Qualitative Analysis

We further analyzed the results of the baseline and two best methods in Figure 4, where metrics sliced by the size of

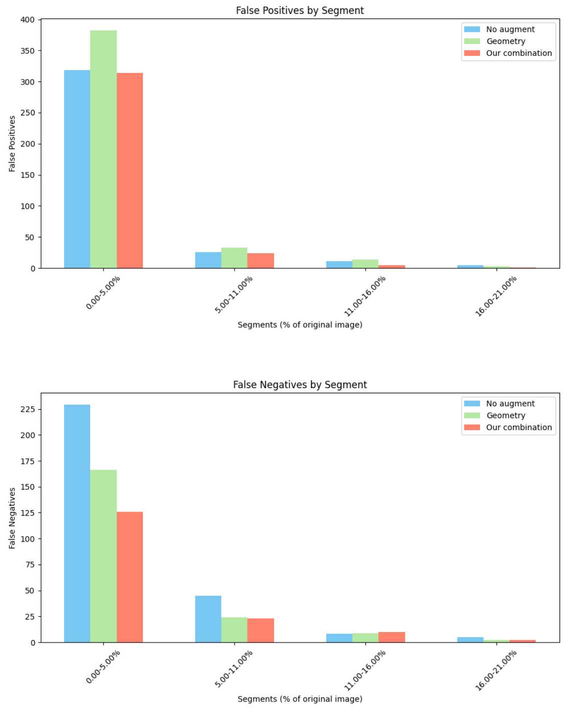


Figure 4. Analysis of false positives and negatives across image size for baseline, geometric, and the proposed full-combo model

Figure 5. Example test results of the baseline and the proposed full-combo method on a typical scene, respectively

false positive and false negative examples are shown. In particular, we provide the results of the geometry augmentation and the proposed full-combo model, comparing them to the no-augmentation baseline. We can see that for false positives the improvement of the proposed approach is notable but not large. However, when it comes to false negatives the proposed method achieved substantially better results, providing only half the number of false negatives compared to the baseline and approximately 20% less than the geometry-augmented model. This is also illustrated in Figure 5 showing a common composite environment within blueberry orchards, along with typical detection results of the baseline and the proposed approaches. We can see that the model trained with the proposed augmentation successfully identified all bushes, unlike the baseline that struggled with smaller objects. Such improvements can be attributed to the new training data used, which introduced complex scenes and transformations that enabled the model to generalize more effectively across all object sizes.

5. Conclusion

We investigated various augmentation techniques for agricultural applications, specifically focusing on blueberry bush detection. We considered and discussed several relevant image transformations, conducting detailed experiments and identifying the augmentation techniques that most impacted the performance of the object detector model. This led us to propose an augmentation approach that combined the approaches that showed the largest metrics improvements, resulting in a comprehensive enhancement of the model's overall effectiveness. This research underscored the necessity of choosing augmentation strategies that are specifically suited to address distinct challenges presented by the problem domain under investigation.

6. Acknowledgements

This research was funded by the ANTARES project under the European Union's Horizon 2020 research and innovation program (SGA-CSA no. 739570 under FPA no. 664387).

References

[1] Massimiliano Agovino, Mariaconcetta Casaccia, Mariateresa Ciommi, Maria Ferrara, and Katia Marchesano. Agriculture, climate change and sustainability: The case of eu-28. *Ecological Indicators*, 105:525–543, 2019. 1

[2] Sonia S Anand, Corinna Hawkes, Russell J De Souza, Andrew Mente, Mahshid Dehghan, Rachel Nugent, Michael A Zulyniak, Tony Weis, Adam M Bernstein, Ronald M Krauss, et al. Food consumption and its impact on cardiovascular disease: importance of solutions focused on the globalized

food system: a report from the workshop convened by the world heart federation. *Journal of the American College of Cardiology*, 66(14):1590–1614, 2015. 1

[3] Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*, 2017. 2

[4] B Sathya Bama, S Mohana Valli, S Raju, and V Abhai Kumar. Content based leaf image retrieval (cblir) using shape, color and texture features. *Indian Journal of Computer Science and Engineering*, 2(2):202–211, 2011. 2

[5] Fardad Dadboud, Vaibhav Patel, Varun Mehta, Miodrag Bolic, and Iraj Mantegh. Single-stage uav detection and classification with yolov5: Mosaic data augmentation and panet. In *2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–8. IEEE, 2021. 2

[6] Environmental Big Data. Performance evaluation of strawberry flower detection by yolov5 and yolov8. *Performance Evaluation*, 240:5. 6

[7] Maarten Elferink and Florian Schierhorn. Global demand for food is rising. can we meet it. *Harvard business review*, 7(04):2016, 2016. 1

[8] Mulham Fawakherji, Ciro Potena, Ibis Prevedello, Alberto Pretto, Domenico D Bloisi, and Daniele Nardi. Data augmentation using gans for crop/weed segmentation in precision farming. In *2020 IEEE conference on control technology and applications (CCTA)*, pages 279–284. IEEE, 2020. 2

[9] Vladan Filipović, Dimitrije Stefanović, Nina Pajević, Željana Grbović, Nemanja Djuric, and Marko Panić. Bush detection for vision-based ugv guidance in blueberry orchards: Data set and methods. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3646–3655, 2023. 2, 5, 6

[10] Jingyao Gai, Lie Tang, and Brian L Steward. Automated crop plant detection based on the fusion of color and depth images for robotic weed control. *Journal of Field Robotics*, 37(1):35–52, 2020. 2

[11] Pablo Gonzalez-De-Santos, Roemi Fernández, Delia Sepúlveda, Eduardo Navas, and Manuel Armada. Unmanned ground vehicles for smart farms. *Agron.-Clim. Chang. Food Secur*, 6:73, 2020. 2

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 5, 11

[13] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *IEEE intelligent systems*, 24(2):8–12, 2009. 2

[14] Sunil S Harakannanavar, Jayashri M Rudagi, Veena I Puranikmath, Ayesha Siddiqua, and R Pramodhini. Plant leaf disease detection using computer vision and machine learning algorithms. *Global Transitions Proceedings*, 3(1):305–310, 2022. 1

[15] Sebastian Haug and Jörn Ostermann. A crop/weed field image dataset for the evaluation of computer vision based precision agriculture tasks. In *Computer Vision-ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part IV 13*, pages 105–116. Springer, 2015. 1

[16] Yiqi Huang, Ruqi Li, Xiaotong Wei, Zhen Wang, Tianbei Ge, and Xi Qiao. Evaluating data augmentation effects on the recognition of sugarcane leaf spot. *Agriculture*, 12(12): 1997, 2022. 2

[17] Muhammad Hussain. Yolo-v1 to yolo-v8, the rise of yolo and its complementary nature toward digital manufacturing and industrial defect detection. *Machines*, 11(7):677, 2023. 6

[18] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, 2023. 6

[19] Andreas Kamilaris and Francesc X Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and electronics in agriculture*, 147:70–90, 2018. 2

[20] Parvinder Kaur, Baljit Singh Khehra, and Er Bhupinder Singh Mavi. Data augmentation for object detection: A review. In *2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 537–543. IEEE, 2021. 2

[21] Phan Truong Khanh, Tran Thi Hong Ngoc, and Sabyasachi Pramanik. Future of smart agriculture techniques and applications. In *Handbook of Research on AI-Equipped IoT Applications in High-Tech Agriculture*, pages 365–378. IGI Global, 2023. 1

[22] Anthony King. Technology: The future of agriculture. *Nature*, 544(7651):S21–S23, 2017. 2

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[24] Bin Liu, Cheng Tan, Shuqin Li, Jinrong He, and Hongyan Wang. A data augmentation method based on generative adversarial networks for grape leaf disease identification. *IEEE Access*, 8:102188–102198, 2020. 2

[25] Yuzhen Lu, Dong Chen, Ebenezer Olaniyi, and Yanbo Huang. Generative adversarial networks (gans) for image augmentation in agriculture: A systematic review. *Computers and Electronics in Agriculture*, 200:107208, 2022. 1, 2

[26] Gurdeep Singh Malhi, Manpreet Kaur, and Prashant Kaushik. Impact of climate change on agriculture and its mitigation strategies: A review. *Sustainability*, 13(3):1318, 2021. 1

[27] Agnieszka Mikołajczyk and Michał Grochowski. Data augmentation for improving deep learning in image classification problem. In *2018 international interdisciplinary PhD workshop (IIPhDW)*, pages 117–122. IEEE, 2018. 1

[28] Daniel Mas Montserrat, Qian Lin, Jan Allebach, and Edward J Delp. Training object detection and recognition cnn models using data augmentation. *Electronic Imaging*, 29: 27–36, 2017. 2

[29] Diego Inácio Patrício and Rafael Rieder. Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and electronics in agriculture*, 153:69–81, 2018. 1

[30] Pornntiwa Pawara, Emmanuel Okafor, Lambert Schomaker, and Marco Wiering. Data augmentation for plant classification. In *Advanced Concepts for Intelligent Vision Systems:*

18th International Conference, ACIVS 2017, Antwerp, Belgium, September 18-21, 2017, Proceedings 18, pages 615–626. Springer, 2017. 2

[31] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017. 2

[32] David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020. 6

[33] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 11

[34] Umme Fawzia Rahim and Hiroshi Mineno. Data augmentation method for strawberry flower detection in non-structured environment using convolutional object detection networks. *Journal of Agricultural and Crop Research*, 8(11):260–271, 2020. 2

[35] Maryam Rahnemoonfar and Clay Sheppard. Deep count: fruit counting based on deep simulated learning. *Sensors*, 17 (4):905, 2017. 1

[36] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 6

[37] Cynthia Rosenzweig and Martin L Parry. Potential impact of climate change on world food supply. *Nature*, 367(6459): 133–138, 1994. 1

[38] Chenglei Si, Zhengyan Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. Better robustness by more coverage: Adversarial and mixup data augmentation for robust finetuning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1569–1576, 2021. 4

[39] Juan Terven and Diana Cordova-Esparza. A comprehensive review of yolo: From yolov1 to yolov8 and beyond. *arXiv preprint arXiv:2304.00501*, 2023. 6

[40] Omer Wosner, Guy Farjon, and Aharon Bar-Hillel. Object detection in agricultural contexts: A multiple resolution benchmark and comparison to human. *Computers and Electronics in Agriculture*, 189:106404, 2021. 1

[41] Qiufeng Wu, Yiping Chen, and Jun Meng. Dcgan-based data augmentation for tomato leaf disease identification. *IEEE access*, 8:98716–98728, 2020. 2

[42] Bingjie Xiao, Minh Nguyen, and Wei Qi Yan. Fruit ripeness identification using yolov8 model. *Multimedia Tools and Applications*, pages 1–18, 2023. 6

[43] Guoliang Yang, Jixiang Wang, Ziling Nie, Hao Yang, and Shuaiying Yu. A lightweight yolov8 tomato detection algorithm combining feature enhancement and attention. *Agronomy*, 13(7):1824, 2023. 6

[44] Linjun Zhang, Zhun Deng, Kenji Kawaguchi, Amirata Ghorbani, and James Zou. How does mixup help with robustness and generalization? *arXiv preprint arXiv:2010.04819*, 2020. 4

[45] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023. 6

A. Appendix

A.1. Generative Model

In our study, we began with a standard Deep Convolutional Generative Adversarial Network (DCGAN) [33] framework as our foundation and expanded its layers to better accommodate images of size 3x128x128 instead of the conventional 3x64x64. This enhancement was designed to more accurately capture the intricacies and details of the higher-resolution images we seek to replicate, thereby improving the model's proficiency in generating synthetic images that closely mirror the real bush photographs quality and texture.

The architecture of our refined DCGAN includes two main components: a generator (G) and a discriminator (D). The generator's objective is to fabricate images that are indiscernible from real images, beginning from a latent space vector z, sampled from a standard normal distribution. Conversely, the discriminator evaluates images to determine their authenticity, real (from the dataset) or fake (generated by G), outputting a probability score.

Our generator is constructed with convolutional-transpose layers, batch normalization layers, and ReLU activations, designed to transform a latent vector into a 3x128x128 RGB image. In line with the DCGAN paper's recommendations, all model weights are initialized from a normal distribution with mean=0 and standard deviation=0.02, to promote effective training dynamics.

```
# input is ``(nc) x 128 x 128``
nn.Conv2d(nc, ndf // 2, 4, 2, 1, bias=False),
nn.LeakyReLU(0.2, inplace=True),
# state size. ``(ndf//2) x 64 x 64``
nn.Conv2d(ndf // 2, ndf, 4, 2, 1, bias=False),
nn.BatchNorm2d(ndf),
nn.LeakyReLU(0.2, inplace=True),
# state size. ``(ndf) x 32 x 32``
nn.Conv2d(ndf, ndf * 2, 4, 2, 1, bias=False),
nn.BatchNorm2d(ndf * 2),
nn.LeakyReLU(0.2, inplace=True),
# state size. ``(ndf*2) x 16 x 16``
nn.Conv2d(ndf * 2, ndf * 4, 4, 2, 1, bias=False),
nn.BatchNorm2d(ndf * 4),
nn.LeakyReLU(0.2, inplace=True),
# state size. ``(ndf*4) x 8 x 8``
nn.Conv2d(ndf * 4, ndf * 8, 4, 2, 1, bias=False),
nn.BatchNorm2d(ndf * 8),
nn.LeakyReLU(0.2, inplace=True),
# state size. ``(ndf*8) x 4 x 4``
nn.Conv2d(ndf * 8, 1, 4, 1, 0, bias=False),
nn.Sigmoid(),
```

Listing 1. Generator code

The discriminator is composed of strided convolution layers, batch normalization layers, and LeakyReLU activations. It processes input images of size 3x128x128, classifying them as real or fake. Importantly, to avoid the discriminator from becoming overly confident in its assessments—a scenario that could significantly impede the generator's learning gradient—we opted to employ soft labels rather than hard binary labels. Specifically, we adjusted the labels for the real images to 0.8 (instead of 1.0) and for the fake images to 0.2 (instead of 0.0). This modification ensures that the discriminator's output probabilities are never absolute (i.e., 100% real or fake), fostering a more nuanced and continuous learning environment for both components of the DCGAN. By implementing soft labels at these levels, we strike a balance that encourages the discriminator to maintain a level of uncertainty, thereby preventing it from dominating the generator too early in the training process and ensuring a healthier gradient flow for the generator's ongoing learning. The optimization of both the generator and discriminator is performed using the Adam optimizer, with a learning rate of 0.0002 and Beta1 set to 0.5.

```
# input is Z, going into a convolution
nn.ConvTranspose2d(nz, ngf * 8, 4, 1, 0, bias=
    False),
nn.BatchNorm2d(ngf * 8),
nn.ReLU(True),
# state size. ``(ngf*8) x 4 x 4``
nn.ConvTranspose2d(ngf * 8, ngf * 4, 4, 2, 1,
    bias=False),
nn.BatchNorm2d(ngf * 4),
nn.ReLU(True),
# state size. ``(ngf*4) x 8 x 8``
nn.ConvTranspose2d(ngf * 4, ngf * 2, 4, 2, 1,
    bias=False),
nn.BatchNorm2d(ngf * 2),
nn.ReLU(True),
# state size. ``(ngf*2) x 16 x 16``
nn.ConvTranspose2d(ngf * 2, ngf, 4, 2, 1, bias=
    False),
nn.BatchNorm2d(ngf),
nn.ReLU(True),
# state size. ``(ngf) x 32 x 32``
nn.ConvTranspose2d(ngf, ngf // 2, 4, 2, 1, bias=
    False),
nn.BatchNorm2d(ngf // 2),
nn.ReLU(True),
# state size. ``(ngf//2) x 64 x 64``
nn.ConvTranspose2d(ngf // 2, nc, 4, 2, 1, bias=
    False),
nn.Tanh(),
```

Listing 2. Discriminator code

A crucial aspect of training involves the loss functions for both G and D, utilizing the Binary Cross Entropy (BCE) loss, expressed as:

$$\begin{aligned} \ell(x, y) = L = \{l_1, \dots, l_N\}^\top, \\ l_n = -\left[y_n \cdot \log(x_n) + (1 - y_n) \cdot \log(1 - x_n)\right] \end{aligned} \tag{1}$$

This loss function calculates both components of the objective function, $\log(D(x))$ and $\log(1 - D(G(z)))$, allowing for specific parts of the BCE equation to be utilized depending on the input. The GAN loss function, pivotal to the training of both G and D in their min-max game, is given by [12]:

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

In this min-max game, D aims to maximize the probability of correctly classifying real and fake images, while G strives to minimize the probability of D identifying its outputs as fake. The theoretical equilibrium of this game is achieved when $p_g = p_{\text{data}}$, and D classifies the inputs as real or fake with equal probability. However, in practice, achieving this equilibrium remains a complex challenge, with the convergence theory of GANs still under active investigation.