

Article

Conditional Generative Adversarial Networks and Deep Learning Data Augmentation: A Multi-Perspective Data-Driven Survey Across Multiple Application Fields and Classification Architectures

Lucas C. Ribas * , Wallace Casaca  and Ricardo T. Fares 

São Paulo State University (UNESP), Institute of Biosciences, Humanities and Exact Sciences (IBILCE),

São José do Rio Preto 15054-000, SP, Brazil; wallace.casaca@unesp.br (W.C.); rt.fares@unesp.br (R.T.F.)

* Correspondence: lucas.ribas@unesp.br

Abstract: Effectively training deep learning models relies heavily on large datasets, as insufficient instances can hinder model generalization. A simple yet effective way to address this is by applying modern deep learning augmentation methods, as they synthesize new data matching the input distribution while preserving the semantic content. While these methods produce realistic samples, important issues persist concerning how well they generalize across different classification architectures and their overall impact in accuracy improvement. Furthermore, the relationship between dataset size and model accuracy, as well as the determination of an optimal augmentation level, remains an open question in the field. Aiming to address these challenges, in this paper, we investigate the effectiveness of eight data augmentation methods—StyleGAN3, DCGAN, SAGAN, RandAugment, Random Erasing, AutoAugment, TrivialAugment and AugMix—throughout several classification networks of varying depth: ResNet18, ConvNeXt-Nano, DenseNet121 and InceptionResNetV2. By comparing their performance on diverse datasets from leaf textures, medical imaging and remote sensing, we assess which methods offer superior accuracy and generalization capability in training models with no pre-trained weights. Our findings indicate that deep learning data augmentation is an effective tool for dealing with small datasets, achieving accuracy gains of up to 17%.



Academic Editor: Arslan Munir

Received: 11 November 2024

Revised: 10 January 2025

Accepted: 5 February 2025

Published: 7 February 2025

Citation: Ribas, L.C.; Casaca, W.; Fares, R.T. Conditional Generative Adversarial Networks and Deep Learning Data Augmentation: A Multi-Perspective Data-Driven Survey Across Multiple Application Fields and Classification Architectures. *AI 2025*, *6*, 32. <https://doi.org/10.3390/ai6020032>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: conditional generative adversarial networks; data augmentation; deep learning; deep neural networks; image processing

1. Introduction

In the rapidly evolving field of Artificial Intelligence, Deep Learning Networks (DLNs) have become the core approach for solving complex pattern recognition and computer vision problems, including medical diagnosis [1,2], face recognition [3,4], image segmentation [5,6] and object detection [7,8]. Particularly, DLNs play an important role in many of these applications, achieving exceptional results while capturing multifaceted patterns from large datasets. Such performance is attributed to their learning process, which preserves the local spatial relationships and creates useful low-dimensional representations.

Despite their good performance in various contexts, training deep learning models typically requires large-scale datasets. A lack of representative instances during the training stage may hamper model generalization, mainly because the trained models tend to overfit the limited amount of training data [9,10]. This becomes a very challenging task, especially

in fields with absence of data, such as medicine, agriculture and botany [9,11,12]. To overcome this issue, many functional approaches have been proposed in the literature, including Dropout [13], Batch Normalization [14], Batch Renormalization [15], Transfer Learning [16,17] and Pre-Training [18,19]. However, the effectiveness of these approaches may vary based on the network complexity and the diversity of the dataset, as they still struggle to prevent overfitting [20]. In such cases, data augmentation stands out as an effective solution, expanding the training dataset and enriching the model with diverse examples, which, in turn, enhances its ability to generalize to new, unseen data.

Traditional data augmentation approaches, such as geometric transformations (e.g., translation, rotation, flipping, cropping) [21], color adjustments (e.g., brightness, contrast, saturation and hue) [22], kernel filters (e.g., Gaussian blur) and image mixing [23], can be used to expand the dataset, achieving desirable outcomes in certain domains like medical diagnosis [24] and plant leaf disease recognition [25]. However, conventional data augmentation methods pose limitations in the diversity and complexity of the data they generate. The modifications made to the input images may retain similar textural visual cues, failing to introduce new features. As a result, this leads to only marginal improvements in quality, as the synthesized data lack the novelty necessary to enhance model generalization and robustness. Another issue pertains to the hyperparameters of these techniques, particularly concerning the geometric position, rotation and color information of the images, which affect the semantic content [26]. Essentially, the data augmentation process should not cause modifications in the original image labels. For example, if an image displays the number 6, rotating it by 180 degrees geometrically transforms it into the number 9, thus altering its semantic content, as illustrated in Figure 1.

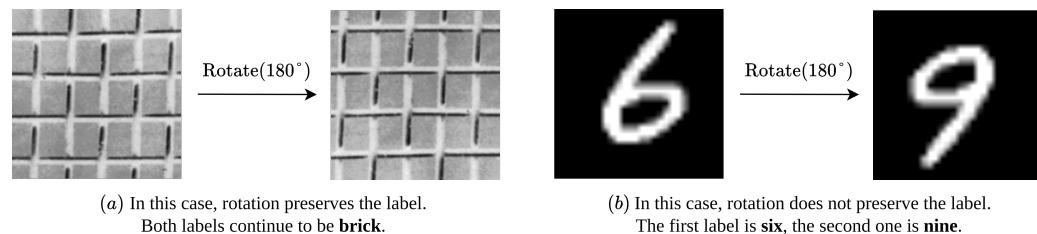


Figure 1. Example of rotation geometric transformation. (a) Rotation does preserve label. (b) Rotation does not preserve label.

More recently, deep learning-based augmentation methods have proven to be effective in overcoming most of the previously mentioned issues. By employing sophisticated architectures such as Generative Adversarial Networks (GANs) [27], these methods can generate high-quality synthetic data, introducing a greater diversity of features while keeping the original labels. Unlike conventional augmentation approaches that rely on geometric transformations or kernel filters, deep learning-based methods are designed to capture subtle variations within the data, leading to more realistic and varied training samples that mimic real-world examples. This advancement improves the model's capacity for generalization, particularly in domains where data scarcity is a critical issue. Additionally, the use of deep learning for augmentation purposes enables more controlled hyperparameter fine-tuning, ensuring that the semantic content of the images remains intact [28,29].

Although deep learning-based data augmentation methods have resulted in notable improvements in certain application contexts, they have been predominantly tested on a small number of classification networks and experimental datasets [30]. Indeed, most existing works on data augmentation focus on narrow contexts, limiting their scope and hindering a broader assessment of how these methods perform across different datasets and architectures, ranging from shallow to deep networks. This lack of comprehensive evaluation raises questions about which deep augmentation methods are more effective in

terms of accuracy, robustness and generalizability for a broader range of available datasets and neural network structures. Exploring how small datasets can be effectively scaled to enhance the accuracy of classification methods leads to another intriguing question: is there an optimal level of expansion for the various popular deep augmentation methods? Finally, given the variability encountered in real-world scenarios, assessing how deep learning augmentation techniques affect model generalization across diverse applications and datasets is of paramount importance for readers who aim to apply these insights to improve model performance in their specific fields.

Aiming at dealing with the above-discussed issues, in this paper, we investigate the effectiveness of eight modern deep learning data augmentation methods in improving the generalization of four well-established classification neural networks. More specifically, we assess the accuracy performance of StyleGAN3 [31], DCGAN [32], SAGAN [33], RandAugment [34], Random Erasing [35], AutoAugment [36], TrivialAugment [37] and AugMix [23], while also testing five augmentation factors across four state-of-the-art deep classification networks: ResNet18 [38] and ConvNeXt-Nano [39], with shallow architectures, and DenseNet121 [40] and InceptionResNetV2 [41], with deeper architectures. In all experiments, we used these techniques in their standard forms without modifications to assess their effectiveness and impact on performance. We also compare three representative datasets from varied application domains, including leaf textures, medical imaging and remote sensing scenes, to provide a thorough analysis of all deep augmentation techniques, highlighting their strengths and limitations.

In summary, the main contributions of this paper are the following:

- The design of a comprehensive pipeline for analyzing data augmentation methods intended to assist readers in implementing these techniques in their own research or industry projects.
- A thorough experiment-based analysis of both conventional deep learning and deep generative data augmentation methods, assessing their effectiveness when applied to shallow as well as deep classification architectures.
- An in-depth analysis of the strengths and limitations of data augmentation methods, offering practical guidance across three real-world image-based applications.

This paper is outlined as follows: In Section 2, we describe the related work using conventional and deep generative data augmentation approaches for solving poor model generalization. In Section 3, we describe the materials and methods, i.e., the datasets used for comparison, the data augmentation pipeline and the data augmentation and classification approaches compared in our analysis. In Section 4, we present the experimental setup as well as our results, findings, and discussion. Finally, the conclusions are presented in Section 5.

2. Related Work

In this section, we discuss different data augmentation techniques with a focus on those that recently employ generative adversarial networks for data generation, alongside comparative studies that highlight the advantages and limitations of the augmentation approaches. It is noteworthy that several works have incorporated data augmentation in image classification and segmentation tasks, leading to different conclusions. These outcomes strongly depend on factors such as the choice of deep learning models, as well as the distribution of data and quality of the instances.

This section is structured as follows: Section 2.1 discusses conventional methods commonly found in the specialized literature, including geometric transformations and noise injection. Section 2.2 reviews the application of GANs for generating realistic data, while Section 2.3 covers hybrid and domain-specific methods for data augmentation.

2.1. Data Augmentation Using Conventional Approaches

Over the last few years, data augmentation techniques have gained considerable traction across various fields of application. A notable contribution is the work proposed by Srinivasu et al. [42], which achieved outstanding classifications of skin diseases by combining popular data augmentation techniques with deep learning strategies to improve data imbalance. The data augmentation task was pivotal in their work as it enlarged and diversified the datasets through variations in image orientation and scale, addressing key challenges, particularly the overfitting issue caused by inherent class imbalance.

The work by Zhang et al. [43] presented a new data augmentation strategy named Two-Stage Random RandAugment (TRRA), which improved the so-called RandAugment method, initially proposed by Cubuk et al. [34] for classifying Alzheimer's disease from structural magnetic resonance images (sMRIs). TRRA incorporates 23 geometric and color transformations, such as flipping, shearing, scaling, noise injection, solarizing, among others. These transformations are randomly selected and divided into two distinct groups: the first category consists of geometric transformations, while the second one is composed of color transformations. Each transformation within these groups has a given probability P of being applied to an image. Their research hypothesized that TRRA not only increases the diversity and size of the training dataset but also significantly improves classification accuracy. Moreover, their findings demonstrated remarkable classification accuracies, outperforming many other well-established methods.

Raj et al. [44] introduced a crossover-inspired data augmentation technique to cope with the issue of limited medical data when training convolutional neural networks for medical image classification. Their technique involves randomly selecting two images from the training dataset and swapping portions of their rows and columns to obtain two new samples. Although their approach does not preserve labels, the experimental results led to a 3.57% improvement in accuracy for skin cancer image classification, alongside a straightforward data augmentation technique. In another relevant work from the medical field, Anwar et al. [45] investigated the impact of data augmentation on electrocardiography (ECG) signals for detection of COVID-19 and heart diseases. Based on their extensive experiments, the authors concluded that conventional data augmentation techniques do not seem to yield significant improvements.

2.2. Data Augmentation Using Generative Adversarial Networks

The works discussed above rely on conventional data augmentation techniques or combinations of them; none of which employ deep learning-based methods for generating new data instances. We now discuss deep learning approaches that specifically use generative adversarial networks (GANs) for data augmentation.

In [46], Min et al. proposed a data augmentation approach based on an image-to-image translation model, integrating two generators and two discriminators. By using cycle consistency loss, their method performs translations from healthy to diseased leaf images, drawing inspiration from the popular CycleGAN architecture [47]. Additionally, their study incorporated attention mechanisms to enhance the poor image conversion outputs of the translation. The experimental results demonstrated that their method improved accuracy in plant leaf disease recognition while achieving lower Fréchet Inception Distance (FID) scores compared to the CycleGAN architecture.

Another recent GAN-based approach is the one developed by Branikas et al. [48], where a data augmentation technique based on a cycle-consistency GAN was taken to generate samples for crack detection datasets, primarily aimed at nuclear plant inspection. The goal of such an approach was to enhance the segmentation accuracy of deep learning models for detecting surface cracks in structural monitoring. Moreover, their approach

eliminates the need for manual annotation of samples, a process that is both labor-intensive and time-consuming. Lastly, the experimental results demonstrated that images generated by their GAN significantly improved the segmentation task.

Aiming to tackle the scarcity of abnormal images, Liu et al. [49] introduced AnomalyGAN, a generative adversarial network specifically designed to expand datasets for surface anomaly detection. This network took a combination of a mask pool, abnormal wear loss and local versus global discriminators to create realistic images. Their evaluation results demonstrated a lower Fréchet Inception Distance score, along with significant increases of 25.6% in mean Average Precision and 24.2% in mean Intersection Over Union.

Conditional Generative Adversarial Networks (cGANs) are increasingly used for their ability to condition both the generator and discriminator on class-specific information, enabling efficient multi-class image generation within a single network. For instance, Bird et al. [50] demonstrated the versatility of cGANs by generating synthetic images of fruits, producing both healthy specimens and those with defects like mold and gangrene. Zhou et al. [51] assessed geometric and color transformations, along with StyleGAN-based augmentation, for sewer defect detection using the YOLO network, finding that geometric transformations outperformed color ones, while StyleGAN enhanced data quality and results. Su et al. [52] introduced a pre-trained StyleGAN-based augmentation method to transfer knowledge from MRI to CT domains, improving CT motion artifact detection and classification sensitivity while tackling the small sample issue in medical imaging. While effective in their respective domains, these studies lack an in-depth evaluation of key factors, such as data expansion levels, or a comparative analysis of multiple augmentation methods across diverse domains—a gap our work seeks to fill.

2.3. Other Works on Data Augmentation Approaches

We finish this section by discussing a few works devoted to comparing data augmentation approaches. These works highlight the benefits and limitations of each approach, providing valuable guidance for future practical applications.

Nanni et al. [53] conducted a comparative study of various data augmentation approaches, including two proposed methods: discrete wavelet transform and constant-Q Gabor transform. The study relied on fine-tuning pre-trained strategies, such as the utilization of ResNet50 networks with each augmentation method across four datasets: virus, bark, portrait, and LIGO glitches. The experiments showed improvements across all datasets.

To conclude, Naveed et al. [54] focused on two specific data augmentation approaches: image mixing and deleting. Their work categorized the different characteristics of numerous strategies within these techniques, such as random erasing, MixUp, AugMix and CutMix. Additionally, the authors pointed out the side benefits of these techniques in training deep learning models, including robustness against image corruption.

In this paper, we investigate the effectiveness of several deep generative data augmentation methods in enhancing classification tasks. Unlike previous studies that focus on specific domains and augmentation methods, our work provides a comprehensive evaluation of both well-established and modern deep learning-based techniques, using various datasets from different domains. We also emphasize the application of cGANs, comparing their performance with others in a way not covered in prior research. Lastly, we offer practical insights into how data augmentation-driven strategies impact model generalization in classification, guiding researchers in enhancing their models.

3. Materials and Methods

In this section, we provide an in-depth description of the datasets, data augmentation techniques and experimental setups employed in our analyses. Our goal is to systematically

assess and compare the performance of modern deep learning-based approaches when trained from scratch on diverse datasets, spanning multiple domains such as texture analysis, medical imaging and remote sensing.

3.1. Datasets

In our assessments, three full data collections were taken when tuning and training the augmentation methods: the 1200Tex [55], Kather [56] and Brazilian Coffee Scenes [57] datasets. Each dataset features different domains of applications, resolutions and visual characteristics, enabling us to address a broad spectrum of variations in the dataset samples. Figure 2 illustrates a few samples from the aforementioned datasets, while Table 1 provides a summary of their main aspects.

1200Tex texture dataset [55]: This dataset consists of 1200 samples of leaf textures grouped among 20 distinct classes, with each class containing 60 images, and with each sample being a 128×128 colored image. This is a widely used dataset employed in the context of plant species recognition, posing a significant challenge due to its reliance on leaf textures. Indeed, the identification of species based solely on leaf textures is a complex, non-trivial task mainly because of the high variability in textural patterns, both among leaves of the same species and within different sections of a single leaf. This dataset was split into 800 images for training, 40 for validation, and 360 for test.

Kather medical dataset [56]: This dataset includes 5000 samples derived from hematoxylin and eosin (H&E) staining analyses of ten colorectal cancer (CRC) tissues. Each of the ten tissue samples was cropped so as to produce 625 non-overlapping sections, each measuring 150×150 ($74 \mu\text{m} \times 74 \mu\text{m}$) pixels. In addition, the visual texture features present in these tissues vary in scale from individual cells ($10 \mu\text{m}$) to mucosal glands (greater than $50 \mu\text{m}$), such that each sample has different organizations. This dataset was split into 4480 images for training, 56 for validation, and 464 for test.

Brazilian Coffee Scenes (BCS) remote sensing dataset [57]: This dataset comprises 2876 images split into two classes: coffee and non-coffee. Each class contains 1438 colored images of 64×64 pixels obtained by cropping scenes taken from the SPOT satellite (using green, red and near-infrared bands) in 2005 over four counties in the state of Minas Gerais, Brazil: Arceburgo, Guaranésia, Guaxupé and Monte Santo. Additionally, this dataset poses challenges resulting from intraclass variations which are attributed to the diverse agricultural practices. This dataset was split into 2584 images for training, 64 for validation, and 228 for test.

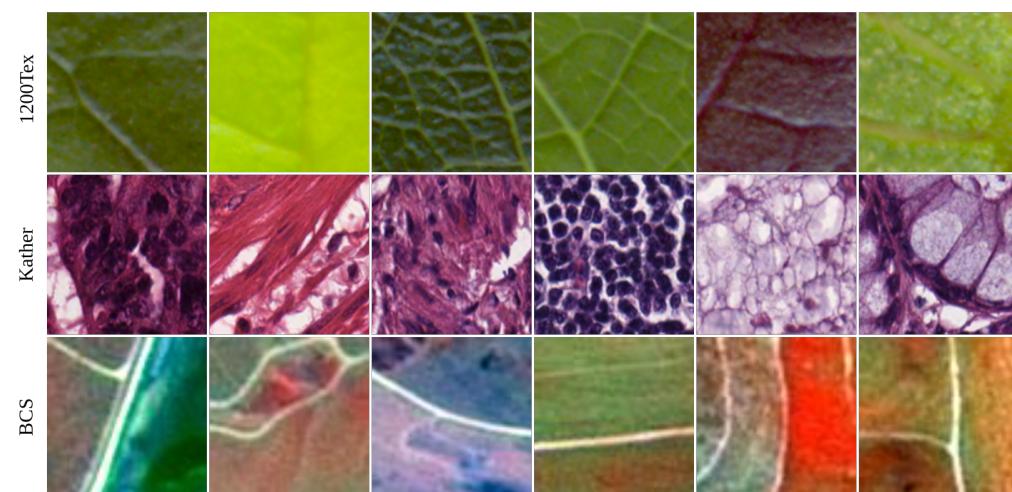


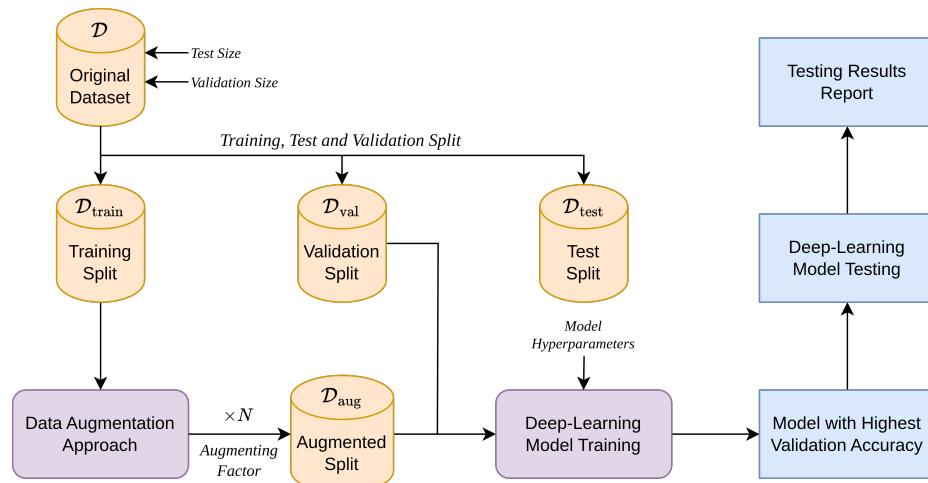
Figure 2. Six samples from the 1200Tex, Kather, and Brazilian Coffee Scenes (BCSs) data collections, with each column representing a sample and each row corresponding to a dataset.

Table 1. Main characteristics of the three datasets used in the experiments.

Dataset	1200Tex [55]	Kather [56]	Brazilian Coffee Scenes [57]
# Samples	1200	5000	2876
# Classes	20 classes	10 classes	2 classes
Image Res.	128 × 128	150 × 150	64 × 64
Image Type	Colored (RGB)	Colored (RGB)	Colored (RGB)
Domain	Plant species recognition	Medical tissue classification	Remote sensing
Challenges	High texture variability	Variability in tissue textures	Intraclass variation (agricultural practices)
Sample Source	Leaf textures of plant species	Hematoxylin and eosin stained tissues	Satellite images from SPOT

3.2. Proposed Learning Methodology

Figure 3 presents the proposed learning pipeline. The initial step involves inputting the target dataset with n instances, $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$, to be augmented. The sizes of the test split T , as well as the validation split V , are also provided. Next, the target dataset is split into training, test and validation sets, denoted here as $\mathcal{D}_{\text{train}}$, $\mathcal{D}_{\text{test}}$ and \mathcal{D}_{val} , respectively, where these sets are mutually exclusive, forming a partition of \mathcal{D} .

**Figure 3.** Schematization of proposed data augmentation training and testing pipeline.

The training split, $\mathcal{D}_{\text{train}}$, serves as the input for the data augmentation process. The selected data augmentation approach and augmentation factor N are applied to expand the size of the training dataset by a factor of N (e.g., if the training split size is 100 images and the augmentation factor is 5, then the augmented split will contain 500 images). Following this approach, the augmented split \mathcal{D}_{aug} is employed to train a specified deep learning model. Meanwhile, the validation split is used to calculate the validation accuracy and select the best model, i.e., the one that achieves the highest validation accuracy.

Lastly, once the model with the highest validation accuracy is identified, the test split $\mathcal{D}_{\text{test}}$, which represents the unseen data, is taken to test the trained model. Thus, the accuracy for the specified dataset, augmentation approach, deep learning model and augmentation factor is determined. These results serve as guidance in the analysis of each deep learning model, the data augmentation approach and the factor of augmentation.

To comprehensively compare recent data augmentation techniques, we explored eight approaches, offering a mix of both deep generative and conventional methods. As a representative of cutting-edge GAN-based augmentation, we chose StyleGAN3 [31], while the modern state-of-the-art augmentation techniques Random Erasing [35], Rand Augment [34], Auto Augment [36], Trivial Augment [37] and AugMix [23] were selected for

comparison. We also included DCGAN [32] and SAGAN [33] to further enrich the comparisons. This setup aims at highlighting the differences between GANs and conventional deep learning techniques, especially when tackling complex datasets where data diversity and augmentation quality are essential for enhancing model generalization and performance. The augmented samples from the evaluated approaches are illustrated in Figure 4.

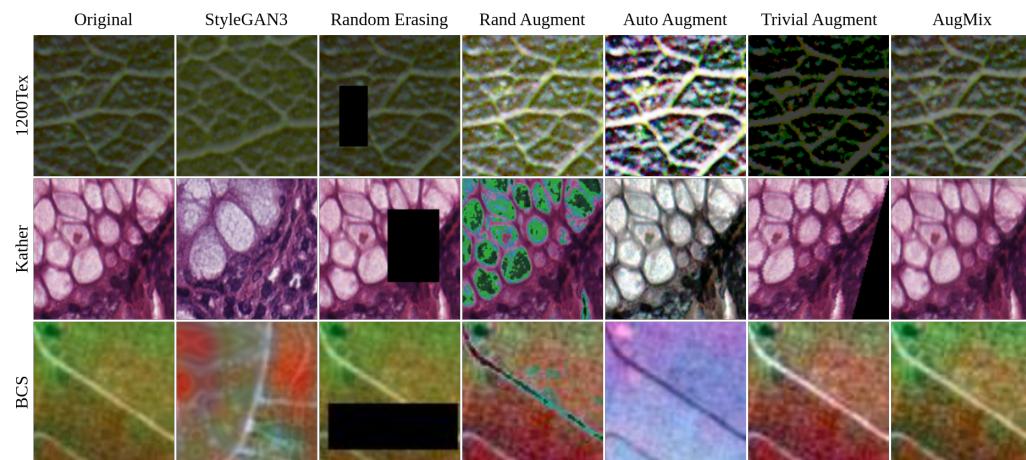


Figure 4. Samples generated by all evaluated data augmentation approaches.

Lastly, we also took four different deep convolutional neural architectures as part of our data augmentation pipeline. More precisely, we selected two lesser depth networks, namely, ResNet18 [38] (18 layers) and ConvNeXt-Nano [39] (14 layers), and two greater depth networks, namely, DenseNet121 [40] (121 layers), and InceptionResNetV2 [41] (164 layers). The use of the same augmentation technique and factor across these models enables us to analyze and compare their behavior and performance within the data augmentation process. We briefly describe uniqueness of each deep neural architecture as follows:

ResNet18: In the early days of AI, training deep neural networks was hindered by vanishing and exploding gradients, partially addressed by normalization techniques. However, the degradation problem, marked by accuracy saturation and decline, persisted. To overcome this, He et al. [38] introduced residual networks (ResNets), which use shortcut connections to stabilize training by enabling direct information and gradient flow. ResNets are available in various depths, such as 18, 34, 50 and 152 layers. In this study, we selected ResNet18.

ConvNeXt-Nano: With the emergence of ViTs, which outperformed traditional CNNs using not-so-image-specific mechanisms, such as attention, efforts to evolve CNNs were initiated. In this context, Liu et al. [39] proposed ConvNeXt, a modernized CNN architecture that incorporates elements inspired by ViTs, such as patchify, layer normalization, inverted bottlenecks, larger kernel sizes and the GELU activation function, among others. This design achieved outstanding results bridging the gap between CNNs and ViTs. Finally, similar to ResNets, ConvNeXt models are available in various configurations with different depths, including Nano, Tiny, Base, Large and X-Large. Here, we selected the ConvNeXt-Nano.

DenseNet121: Building on the idea that convolutional neural networks can be made deeper by using skip connections, as demonstrated by ResNets, Huang et al. [40] proposed DenseNet, a network characterized by its dense connectivity, where each layer is connected to all previous layers. This dense connectivity enhances training stability by mitigating vanishing and exploding gradient issues. It also improves the learning of distinguishing image features, as each layer can access the feature maps of all preceding layers. This promotes feature reuse and enables the model to generate better representations.

InceptionResNetV2: Szegedy, Ioffe and Vanhoucke [58] proposed the InceptionResNetV2 architecture, which combines the strengths of the Inception and ResNet models. The Inception model achieves computational efficiency and strong performance by using parallel branches with distinct kernel sizes (1×1 , 3×3 and 5×5), enabling the network to capture multi-scale features effectively. On the other hand, as aforementioned, ResNet introduces skip connections that address the degradation problem. Therefore, as InceptionResNetV2 integrates these two approaches, it leverages the benefits of both architectures.

3.3. Popular Deep Learning Data Augmentation Approaches

This section presents the deep learning data augmentation techniques tested in our data-driven exploration, including their potential benefits during the training stage. Such an analysis aims to provide insights into how these techniques enhance model performance and contribute to the robustness and better generalization of the training process.

Random Erasing is an effective data augmentation method introduced in 2020 by Zhong et al. [35] that randomly erases rectangular regions within the images to simulate occlusions. This erasing scheme is executed with a probability of 0.5, ensuring that the rectangular region remains within the image boundaries. Additionally, the Random Erasing can be effectively combined with other data augmentation methods to further enhance the augmentation process. One significant advantage of this approach is that it introduces variations in occlusion to the generated images, mitigating the risk of overfitting while enhancing the model's robustness to occlusion [35]. Despite its advantageous properties, the random erasing procedure may remove the semantic content from the image.

AutoAugment is a well-established data augmentation technique proposed in 2019 by Cubuk et al. [36] that eliminates the need for manually searching for the best set of image transformations for use in data augmentation. The search space of the method includes policies composed of sub-policies, each featuring an image transformation T , such as rotation, shearing and contrast, along with the probability P of applying the transformation and its magnitude M . This approach automatically searches for the optimal set of image transformations for data augmentation, offering significant benefits. Although it demonstrates robustness, a potential weakness of the AutoAugment method lies in the cost associated with the search; the larger the search space, the more time the search consumes.

RandAugment is an enhanced approach introduced in 2020 by the same authors, Cubuk et al. [34], to address the costly and complex process of searching for the best image transformations, which escalates training complexity and computational costs. This technique simply applies N image transformations at a specified magnitude M . The primary advantage of this method is the drastic reduction in the search space, reducing time and computational demands. However, a potential drawback is that the randomly selected transformations might not necessarily enhance the accuracy of the model.

TrivialAugment is a straightforward data augmentation technique proposed in 2021 by Muller et al. [37] that samples an image transformation and selects a transformation magnitude for each image independently. Despite its simplicity, this method has achieved outstanding results. The primary advantages of TrivialAugment are its simplicity and low cost. However, a noted weakness, as highlighted in [37], is the requirement for tuning when applied to object detection tasks.

AugMix is a data augmentation technique introduced in 2021 by Hendrycks et al. [23] that applies multiple parallel image transformations. Each branch involves a sequence of transformations that is then mixed with the original image to generate a new sample. This approach allows each branch to perform distinct augmentations, and the sequential

operations increase the diversity of the generated images. Hence, the benefits of this method are its being a simple-to-implement method and the diversity introduced in the images.

3.4. Data Augmentation Using Conditional Generative Adversarial Networks

The generative adversarial network (GAN) [59] is a deep generative-type model that consists of two components: a generator and a discriminator. The generator aims to synthesize real samples to fool the discriminator, while the discriminator tries to distinguish the fake samples from real ones. These networks are iteratively trained in this adversarial approach until the discriminator cannot distinguish between fake and real samples. Mathematically, let $z \in \mathbb{R}^d$ be a random vector sampled from a probability distribution, typically the standard normal distribution $\mathcal{N}(0, I)$, where d represents the latent dimension. The generator acts as a function that takes as input the latent vector z and outputs a synthetic sample, represented by $G(z)$. Conversely, the discriminator is a function that takes as input an image x and outputs a probability of that sample being real, where $D(x) = 0$ is for fake samples and $D(x) = 1$ for real ones.

An important aspect to be noted is that the original GAN formulation does not account for the class of the sample during the generation process, meaning that the class of the generated sample cannot be controlled [60]. Given that controlling the class of the sample is crucial in many contexts of application, particularly for image annotation tasks, in our approach, we implement a *Conditional Generative Adversarial Network* (cGAN). This strategy allows for the generation of samples specific to predetermined classes, thus embedding class information into the generation process. This is essential because, without the conditional mechanism, the generated samples may not produce labels that are consistent with each class, leading to mismatched or unusable data.

The cGAN focuses on modeling the underlying conditional distribution of the original dataset \mathcal{D} using the same adversarial approach as that in the original GAN model. Nevertheless, in this case, a vector $y \in \mathbb{R}^k$, where k indicates the number of classes, is combined with the random latent vector so as to produce the sample $G(z, y)$. During the training stage, the discriminator D aims to maximize the value of $D(x, y)$, where x is a real sample, while minimizing the value of $D(G(z, y), y)$, where $G(z, y)$ is the generated sample. For cGAN, such a maximization problem is formulated by Equation (1). Simultaneously, the generator seeks to deceive the discriminator by minimizing the probability that $D(G(z, y), y)$ is classified as fake, as expressed in Equation (2):

$$\mathcal{L}_D = \mathbb{E}_{x \sim p_{\text{data}}(x|y)} [\log D_{\theta_d}(x, y)] + \mathbb{E}_{z \sim p(z)} [\log(1 - D_{\theta_d}(G_{\theta_g}(z, y), y))], \quad (1)$$

$$\mathcal{L}_G = \mathbb{E}_{z \sim p(z)} [\log(1 - D_{\theta_d}(G_{\theta_g}(z, y), y))]. \quad (2)$$

The above-described adversarial learning process enables both the generator and the discriminator to enhance their abilities in their respective roles, capturing the underlying conditional distribution of the original data, $p(x|y)$. This demonstrates the effectiveness of the adversarial approach in generating realistic samples that adhere to specific conditions. Lastly, the complete loss function for this learning process is outlined below:

$$\mathcal{L}_{\text{GAN}} = \mathbb{E}_{x \sim p_{\text{data}}(x|y)} [\log D_{\theta_d}(x, y)] + \mathbb{E}_{z \sim p(z)} [\log(1 - D_{\theta_d}(G_{\theta_g}(z, y), y))]. \quad (3)$$

In this context, to obtain a set of synthetic images $\{G(z_i, y)\}_{i=1}^N$ for data augmentation, we sample a set of latent vectors $\{z_i\}_{i=1}^N$, where $z_i \sim p(z)$, and input them to the generator function, G , with each $G(z_i, y)$ being a synthetic sample.

Next, we describe the architectural distinctions among different GANs. It is widely recognized that such architectural choices significantly influence image quality, convergence behavior, and, most notably, computational cost.

DCGAN: The Deep Convolutional Generative Adversarial Network (DCGAN) [32] is a popular GAN architecture that enhances stability and performance by incorporating convolutional layers. Unlike earlier GANs that relied on fully connected layers, DCGAN uses convolutional layers to better capture spatial hierarchies in images. Despite the common challenges in training GANs, DCGAN introduced several strategies to improve both stability and performance, including the following: (i) replacing fully connected layers with global average pooling in the discriminator; (ii) utilizing batch normalization layers in both the generator and discriminator to stabilize learning and reduce mode collapse; (iii) substituting max-pooling with convolutional layers that use larger or fractional strides to preserve spatial information; and (iv) using leaky ReLU activations in the discriminator to prevent vanishing gradients [32,61]. These innovations have solidified DCGAN as a foundational approach in generative models, paving the way for more advanced architectures.

SAGAN: The Self-Attention Generative Adversarial Network (SAGAN) [33] incorporates a self-attention mechanism into convolutional GANs, enabling the modeling of long-range, multi-level dependencies across various regions of the image. The self-attention module enables the generator to produce images with finely coordinated details spanning distant regions, while allowing the discriminator to more effectively enforce geometric consistency across the global image structure. In addition, SAGAN incorporates spectral normalization in the generator—a technique previously applied only to the discriminator, further improving training stability and dynamics [61].

StyleGAN3: To date, there have been three main versions of the StyleGAN (1, 2 and 3) [31,62,63]. Each version has brought specific improvements to the generator architecture of the previous one, enabling the creation of images with progressively higher fidelity. Unlike other architectures, the generator in StyleGAN does not input the sampled latent vector $z \in \mathcal{Z}$ directly into the synthesis network. Instead, z is mapped to an intermediate latent space \mathcal{W} via a non-linear mapping network $f : \mathcal{Z} \rightarrow \mathcal{W}$. Next, the generator applies affine transformations to $f(z) = w \in \mathcal{W}$, producing the styles used throughout the synthesis process. This approach enables greater control over the generation process by introducing a more disentangled latent space \mathcal{W} compared to \mathcal{Z} . Furthermore, to enhance the naturalness and refinement of the generated images, stochastic variations are introduced by injecting per-pixel noise after each convolution step throughout the synthesis process.

However, despite its strong architectural design, it was noted that some images created by the first StyleGAN exhibited blob-like artifacts. The adaptive instance normalization (AdaIN) operation was identified as the cause of this issue, and after addressing it, the artifacts were removed, thereby culminating in StyleGAN2. Nevertheless, in this new version, during interpolation procedures (i.e., generating images along a curve connecting sampled latent vectors z_1 and z_2), it was observed that the texture was sticking to its original coordinates, causing a strange-looking feeling in videos or animations. The texture-sticking issue was traced to aliasing, which was introduced and amplified in the synthesis network. To overcome this, the network was modified to incorporate alias-free operations, achieving an equivariant architectural design. This adjustment successfully eliminated the positional references, resolving the texture-sticking problem and resulting in StyleGAN3.

4. Results and Discussion

In this section, we present and discuss the results obtained from an extensive battery of tests involving the data augmentation techniques and datasets previously discussed. We first address the considerations related to the augmentation factor, followed by the training settings as employed during the experiments. Next, we provide a detailed description of the evaluation metrics used in our assessments, followed by the quantitative and qualitative analyses of the generated outputs.

4.1. Experimental Platform

The experiments were conducted on a server equipped with 128 GB RAM, an Intel i9-14900KF processor and a GeForce RTX 4090 24 GB GPU, running on the Ubuntu 22.04 operating system. The deep learning models were implemented using the Python 3.11.9 language and the PyTorch 2.4.0 and Torchvision 0.19.0 libraries. Additionally, the original StyleGAN3 implementation can be found in [64]. The conventional data augmentation methods we ran and tuned are available via the PyTorch 1.10 library.

4.2. Augmentation Factor Analysis

Each data augmentation technique applied to a training dataset was associated with an augmentation factor, denoted here as N . This factor indicates how many samples will be generated w.r.t. the original training dataset size, $|\mathcal{D}_{\text{train}}|$. Specifically, the augmentation factor determines how many times larger the augmented dataset will be compared to the original one. Mathematically, this can be expressed as $|\mathcal{D}_{\text{aug}}| = N|\mathcal{D}_{\text{train}}| = |\mathcal{D}_{\text{train}}| + |\mathcal{D}_{\text{gen}}|$, where \mathcal{D}_{gen} represents the set of generated samples. For instance, if the original training dataset contains 500 samples, and the augmentation factor is $N = 3$, the augmented dataset will have a total of 1500 samples. This includes the original 500 samples plus 1000 new ones generated using the data augmentation technique.

To quantitatively evaluate and compare the data augmentation techniques, five different augmentation factors are taken: $N \in \{2, 3, 4, 5, 6\}$. This involves comparing the baseline results with augmented datasets that are two to six times the size of the original training dataset. The purpose of selecting a variety of augmentation factors is to assess how the data augmentation techniques and deep learning models perform as the size of the augmented dataset increases. Our analysis indicates that although larger augmented datasets typically lead to better performance in deep learning models, the gains tend to plateau after a certain point. This suggests that simply increasing dataset size is insufficient; rather, the quality and relevance of the augmented data become increasingly important for optimizing the deep learning models.

4.3. Training and Augmentation Settings

In this section, we describe the settings and hyperparameters used in the experiments. Specifically, we outline the hyperparameters for the conventional data augmentation approaches, the training of GANs, and the deep learning models. In general, we did not perform hyperparameter tuning. Instead, we took the default settings provided by the method's authors or the Torchvision library. This choice is intended to ensure that our analysis focuses solely on assessing the effectiveness of the data augmentation methods without introducing other influencing factors. Next, we present in detail the settings employed.

Conventional Augmentation Approaches: We employed AutoAugment using the default ImageNet policy of the Torchvision library. This policy refers to the set of augmentations tailored for ImageNet. Following this, RandAugment, TrivialAugment, AugMix and RandomErasing were also selected with their default settings from Torchvision v0.19.0.

GAN-Based Approaches: To train all the conditional GANs, we applied the default settings. Specifically, the Adam optimizer with $\beta_1 = 0$ and $\beta_2 = 0.9$ was taken for both the generator and discriminator networks. The learning rate for the generator was set to $\gamma_G = 2.5 \times 10^{-3}$, and for the discriminator, $\gamma_D = 2 \times 10^{-3}$. Additionally, the dimensions of both the latent space (\mathcal{Z}) and the intermediate latent space (\mathcal{W}) were set to 512. The training process utilized a batch size of 64 and continued until 8480 kimg (an alternative metric used to quantify the amount of training instead of epochs), which represents the number of thousand real images shown to the discriminator [31,62,63]. For sake of clarity, 10,000 kimg means a total of 10 million real images (10,000 images \times 1000) used during training, corresponding to 156,250 iterations, given the batch size of 64.

Deep Learning Models: To train the deep learning models on both the original and augmented training datasets, the Adam optimizer was used with a learning rate set to $\gamma = 1 \times 10^{-3}$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$ as in [65]. In addition, to minimize sources of randomness, all random number generators were fixed so that the deep learning models of the same type were initialized with the same random weights.

4.4. Evaluation Metrics

In our analyses, three evaluation metrics are taken to assess each data augmentation approach associated with a given deep learning model. More precisely, we employ the Fréchet Inception Distance (FID) [66], Inception Score [67] and the deep learning model accuracy, which are described as follows:

- **Fréchet Inception Distance (FID):** This metric gauges a specific layer of an Inception model to map images into a feature space, where the 2-Wasserstein distance is calculated between the distribution of real images, p_{real} , and the one related to the augmented images, p_{aug} . The FID (4) measures how similar these two distributions are, providing an assessment of the quality of the augmented images. FID is expressed mathematically in terms of Equation (4):

$$\text{FID}(p_{\text{real}}, p_{\text{aug}}) = \|\mu_{\text{real}} - \mu_{\text{aug}}\|_2^2 + \text{trace}\left(\Sigma_{\text{real}} + \Sigma_{\text{aug}} - 2\sqrt{\Sigma_{\text{real}}\Sigma_{\text{aug}}}\right), \quad (4)$$

where μ_{real} , μ_{aug} , Σ_{real} and Σ_{aug} are the mean and covariance of the extracted features from both real and augmented distributions. A lower FID is preferred, since it indicates that the distributions are more similar.

- **Inception Score (IS):** This metric evaluates the quality of augmented images by computing a pre-trained Inception v3 model to classify the augmented images to measure the clarity and diversity of the augmented images. IS is mathematically expressed in terms of Equation (5):

$$\text{IS}(p_{\text{aug}}) = \exp(\mathbb{E}_{\mathbf{x}} \text{KL}(p_{\text{aug}}(y|\mathbf{x}) \| p_{\text{aug}}(y))), \quad (5)$$

where p_{aug} accounts for the probability distribution of the augmented dataset. A higher IS is better, indicating more clarity and diversity of the images.

- **Model Accuracy:** This metric is a typical measure used to assess the overall performance of a classification model. Its computation is given by the ratio between the correctly predicted instances over the total number of instances in the dataset, as follows:

$$\text{Accuracy} = \frac{\text{Number of correctly predicted instances}}{\text{Total number of instances}}. \quad (6)$$

4.5. Quantitative Assessment

In our battery of experiments, we first compare popular data augmentation methods with StyleGAN3, and then analyze and contrast StyleGAN3 with other GAN-type models, demonstrating its superiority for data augmentation purposes.

We start by assessing the quality, clarity and diversity of the compared augmentation approaches using the FID and IS values. Table 2 presents the FID and IS values for the six examined data augmentation approaches when applied to the 1200Tex, Kather and Brazilian Coffee Scenes (BCSs) datasets. From the tabulated scores, one can verify that for the 1200Tex dataset, the StyleGAN3 augmentation approach achieved the best (lowest) FID value, while AutoAugment delivered the best (highest) IS. For the Kather dataset, AugMix produced the best FID score, whereas TrivialAugment reached the best IS. Similarly, for the BCS dataset, AugMix attained the best FID, while TrivialAugment achieved the best IS. Nevertheless, as we shall see, an interesting finding is that the augmentation approaches that produce the best FID or IS values do not necessarily result in the highest model accuracies. This may occur because FID and IS are designed to measure the quality and diversity of generated images relative to the original images. However, they do not influence how the model utilizes this quality or diversity to identify more distinguishing features or enhance its generalization, thereby having no such direct connection to accuracy.

Table 2. Comparison of different data augmentation approaches w.r.t. FID and IS. Bold scores indicate best results. \uparrow indicates higher is better, while \downarrow means lower is better.

Method	1200Tex		Kather		BCS	
	FID (\downarrow)	IS (\uparrow)	FID (\downarrow)	IS (\uparrow)	FID (\downarrow)	IS (\uparrow)
StyleGAN3	28.35	3.10	42.97	3.99	32.29	4.39
RandAugment	59.70	2.79	46.62	4.81	32.45	4.35
Random Erasing	47.64	2.65	34.57	3.98	29.69	4.29
Auto Augment	98.03	3.43	64.00	4.86	39.59	4.38
Trivial Augment	46.63	3.25	30.59	4.95	23.71	4.94
AugMix	39.18	3.06	21.90	4.22	12.79	4.33

We now assess the model accuracies for each analyzed dataset. Since the DenseNet121 architecture consistently achieved the highest accuracies across all datasets, albeit with different augmentation approaches, we focus on keeping the DenseNet121 fixed while varying the augmentation factor N . The quantitative results comparing the ResNet18, InceptionResNetV2 and ConvNeXt-Nano networks will be presented later.

Table 3 summarizes the DenseNet121 model accuracies for the 1200Tex dataset, where the baseline accuracy (i.e., without augmentation) is 76.94%. From the listed scores, it can be seen that the StyleGAN3 resulted in the best model and, consequently, the highest performance gain, jumping from 76.94% to 94.17%. While RandAugment and Random Erasing showed a performance reduction for $N = 5$ and 6 , the others did not exhibit such behavior. However, notice that RandAugment was capable of reaching the second-highest accuracy, 93.06%, indicating that increasing the augmentation factor led to significant accuracy gains with less data. This observation also holds true for the other methods, which were improved by artificially extending the dataset through an augmentation strategy. In fact, in the least impactful case, the accuracy gain was a surprising 11.95%. Finally, although StyleGAN3 obtained the highest performance gain (17.23%) with the lowest FID (28.35), AutoAugment delivered the lowest performance gain (11.95%) with the highest IS (3.43).

Table 3. DenseNet121 model accuracies for the 1200Tex dataset when incrementing the augmentation factor N . The baseline is 76.94%. The best results for each N are shown in bold, while the overall best scores are both bolded and underlined.

Method	$\times 2$	$\times 3$	$\times 4$	$\times 5$	$\times 6$	Gain (%)
StyleGAN3	85.00	86.11	88.61	92.22	<u>94.17</u>	<u>17.23</u>
Rand Augment	75.28	85.56	86.67	92.50	87.78	15.56
Random Erasing	90.00	90.56	92.78	93.06	91.39	16.12
Auto Augment	75.28	70.83	82.78	87.50	88.89	11.95
Trivial Augment	80.28	86.94	83.33	90.28	92.50	15.56
AugMix	82.22	88.61	84.17	90.00	92.22	15.28

Table 4 lists the scores of the classification results generated by the DenseNet121 model for the Kather dataset, where the baseline is 90.52% (i.e., without augmentation). By analyzing the results, it was found that the augmentation task yields improvements for the classifiers in most of the data expansions. In terms of comparative performance, the Random Erasing approach with an augmentation factor of $N = 3$ achieved the highest accuracy of 94.18%, resulting in the greatest performance gain of 3.66%. Unlike the 1200Tex dataset, in the Kather database, Random Erasing reached the highest accuracy with a lower augmentation factor, while for the 1200Tex datasets, StyleGAN3 achieved the best outcome with the highest augmentation factor. Although Random Erasing was the best approach, the second-highest accuracy was achieved by StyleGAN3, with 93.97% (just 0.21% lower than the best), using an augmentation factor of $\times 2$, resulting in a 3.45% gain. Moreover, Random Erasing did not achieve the best FID or the best IS.

Table 4. DenseNet121 model accuracies for the Kather dataset when incrementing the augmentation factor N . The baseline is 90.52%. The best results for each N are shown in bold, while the overall best scores are both bolded and underlined.

Method	$\times 2$	$\times 3$	$\times 4$	$\times 5$	$\times 6$	Gain (%)
StyleGAN3	93.97	90.73	91.16	93.32	92.67	3.45
Rand Augment	90.30	88.58	90.95	89.22	87.93	0.43
Random Erasing	90.52	<u>94.18</u>	91.59	92.24	90.73	<u>3.66</u>
Auto Augment	91.59	92.46	93.10	92.24	91.16	2.58
Trivial Augment	86.64	92.46	93.10	90.52	92.24	2.58
AugMix	92.46	92.89	92.24	92.24	89.66	2.37

Next, the results of the data augmentation approaches when the DenseNet121 model is applied to the Brazilian Coffee Scenes dataset are summarized in Table 5, with a baseline accuracy of 92.11% (i.e., without augmentation). It can be seen that StyleGAN3 achieved the highest model accuracy of 95.18% with the lowest augmentation factor of $N = 2$, resulting in a 3.07% improvement over the baseline. This reinforces the point that larger augmentation factors do not necessarily lead to higher accuracies, suggesting there is an optimal augmentation increment. The Random Erasing approach achieved the second-highest accuracy, 93.42%, with an augmentation factor of $N = 5$, marking a 2.19% improvement over the baseline. Additionally, unlike the 1200Tex and Kather datasets, AutoAugment did not result in any performance gains for this dataset, while RandAugment decreased performance by 0.44% compared to the baseline. These suggest that the Brazilian Coffee Scenes dataset is more sensitive to variations in both the augmentation approach and the data expansion factor.

A common trait present in all examined datasets is that the StyleGAN3 and Random Erasing augmentation approaches achieved the highest accuracies. StyleGAN3 attained the best accuracies in the 1200Tex and Brazilian Coffee Scenes datasets, and the second-best

in the Kather medical dataset. Conversely, Random Erasing achieved the best accuracy in the Kather database, and the second-best in the 1200Tex and Brazilian Coffee Scenes collections. This consistency demonstrates the robustness of deep learning data augmentation approaches across diverse datasets, indicating that they are promising choices for applications in other domains where data scarcity may pose an issue or when a pre-trained reference model is not available for fine-tuning purposes.

Table 5. DenseNet121 model accuracies for the Brazilian Coffee Scenes dataset when incrementing the augmentation factor N . The baseline is 92.11%. The best results for each N are shown in bold, while the overall best scores are both bolded and underlined.

Method	$\times 2$	$\times 3$	$\times 4$	$\times 5$	$\times 6$	Gain (%)
StyleGAN3	95.18	92.98	89.47	92.98	89.47	3.07
Rand Augment	91.23	87.28	91.67	89.04	91.67	-0.44
Random Erasing	92.98	94.30	91.23	93.42	92.54	2.19
Auto Augment	91.23	91.23	91.67	92.11	89.04	0.00
Trivial Augment	92.11	89.04	93.42	89.47	91.23	1.31
AugMix	93.86	90.35	92.11	87.72	92.98	1.75

In comparison to other GAN-based approaches, such as DCGAN and SAGAN, StyleGAN3 achieved strong overall results due to its robust architecture and ability to effectively handle small datasets. For instance, on the Kather dataset, DCGAN and SAGAN achieved accuracies of 89.66% ($\times 4$) and 92.03% ($\times 4$), respectively, showing a decline in model performance compared to the StyleGAN3. This behavior suggests lower image quality and reduced diversity in the generated images, as reflected by the low ISs of 1.68 for DCGAN and 2.19 for SAGAN. These limitations can be attributed to the conditioning imposed during GANs training, which constrains the models by requiring the generation of class-specific images, thus necessitating more complex architectures like StyleGAN3. Similar performances were observed on other datasets, including the BCS one, which contains smaller images. In this dataset, DCGAN and SAGAN achieved accuracies of 92.98% ($\times 4$) and 89.04% ($\times 4$), respectively. The assessments indicate that simpler conditional GAN models, like DCGAN, struggle to generate high-quality images, limiting classification model generalization. These findings point to StyleGAN3 as the most suitable GAN-based model for image augmentation.

Regarding the TrivialAugment and AugMix methods, although they did not yield the highest results, both still performed satisfactorily, making them viable options for applications in other domains of interest. On the other hand, the AutoAugment and RandAugment approaches exhibited the highest variability in accuracies for all datasets. For instance, AutoAugment achieved the lowest performance gain in Table 3, while Table 4 emphasizes that RandAugment yielded the lowest performance gain and resulted in a negative performance gain in Table 5.

It is also important to note that performance decreases after a certain level of N for data augmentation techniques. This behavior can be attributed to several factors. First, as N increases, the variability introduced by the augmentation techniques may reach a saturation point, leading to a lack of meaningful diversity in the training data. This can result in model stagnation, where additional augmented data no longer contributes to improved learning. Second, as N grows, a larger proportion of the training dataset is composed of augmented samples relative to the original images (e.g., for $N = 5$, 4 out of 5 parts of the training set are augmented). This reliance on augmented data may cause overfitting to artifacts or noise introduced by the augmentation techniques, deviating from the original image characteristics and hindering generalization to unseen data, thus reducing performance.

For our last quantitative test, we provide in Tables 6–8 the top four results for each dataset, drawn from the various data expansions performed and the deep learning augmentation approaches. In more technical terms, all possible combinations of augmentation methods and expansion factors were exhaustively tested to ensure a comprehensive evaluation. These results were analyzed under the perspective of four well-established neural network architectures: ResNet18 [38], DenseNet121 [40], InceptionResNetV2 [41] and ConvNeXt-Nano [39]. From the tabulated values, it is evident that out of the twelve best results for each network, five were achieved using StyleGAN3, and another five using Random Erasing. Although Random Erasing is not a deep learning-based method, it is surprising that it is highly competitive, achieving similar scores to StyleGAN3. Both augmentation methods demonstrated robustness across different datasets and various neural network architectures. The summarized results also reveal that networks with greater depth consistently achieved higher accuracies (e.g., DenseNet121 achieved the highest accuracies across all datasets). Moreover, networks with smaller depth also performed well (e.g., ResNet18 attained the third-best, second-best, and first/second-best results in the 1200Tex, Kather, and Brazilian Coffee Scenes datasets, respectively). Despite the generally favorable outcomes, we observed that ConvNeXt-Nano did not perform satisfactorily on the 1200Tex dataset, achieving the lowest results across all datasets. This suggests that in this case, the depth of the network may play a significant role in performance.

Table 6. Combination of different neural networks for classification and deep learning augmentation approaches, focusing on the best augmentation factors when applied to the 1200Tex dataset. The bold score indicates the best outcome.

Network	Accuracy (\uparrow)	Augmentation Method	Aug. Factor N
ResNet18	92.78	StyleGAN3	$\times 6$
DenseNet121	94.17	StyleGAN3	$\times 6$
InceptionResNetV2	93.61	Random Erasing	$\times 4$
ConvNeXt-Nano	69.17	Random Erasing	$\times 4$

Table 7. Combination of different neural networks for classification and deep learning augmentation approaches, focusing on the best augmentation factors when applied to the Kather dataset. The bold score indicates the best outcome.

Network	Accuracy (\uparrow)	Augmentation Method	Aug. Factor N
ResNet18	93.75	StyleGAN3	$\times 5$
DenseNet121	94.18	Random Erasing	$\times 3$
InceptionResNetV2	92.03	StyleGAN3	$\times 6$
ConvNeXt-Nano	90.09	Trivial Augment	$\times 5$

Table 8. Combination of different neural networks for classification and deep learning augmentation approaches, focusing on the best augmentation factors when applied to the BCS dataset. The bold scores indicate the best outcomes.

Network	Accuracy (\uparrow)	Augmentation Method	Aug. Factor N
ResNet18	95.18	Random Erasing	$\times 3$
DenseNet121	95.18	StyleGAN3	$\times 2$
InceptionResNetV2	94.74	AugMix	$\times 6$
ConvNeXt-Nano	93.42	Random Erasing	$\times 4$

4.6. Assessment of the Augmented Data Distributions

Next, we explore how the artificially generated data fit the training data distributions, assessing the degree to which the synthetic samples capture key patterns and characteristics.

Figure 5 presents the t-SNE projections of the original training sets for the 1200Tex, Kather and Brazilian Coffee Scenes data collections. The t-SNE projection [68] is a very effective dimensionality reduction tool used to assess the quality of data augmentation methods [69–72]. They allow for a visual comparison of the distribution of the original training set with that of the augmented dataset, revealing how well the augmented samples capture the characteristics of the original input data.

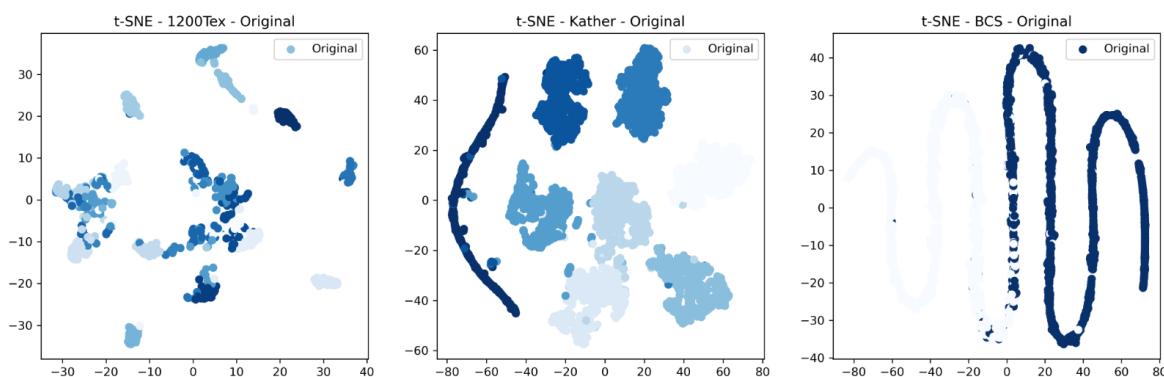


Figure 5. The t-SNE projection of the original training sets for the 1200Tex, Kather and Brazilian Coffee Scenes datasets.

We start by visually inspecting the t-SNE distributions plotted in Figure 6, which corresponds to the leaf texture dataset. It becomes apparent that StyleGAN3 appears to most closely resemble the original distribution, which is expected since its training generative mechanism is specifically designed to make the probability distribution of the generated data match the real one. Random Erasing also promotes a notable similarity between its outputs and the input distribution, likely because it only erases parts of the image without altering other pixels. AutoAugment maintains some similarity to the original projected data, though its diverse transformations result in slight deviations. RandAugment and TrivialAugment, on the other hand, show more scattered points, producing a less precise resemblance. Finally, AugMix attempts to balance diversity with some alignment to the input data, but it also introduces artificial clusters in the lower right corner.

Figure 7 displays the t-SNE visualization of the six deep learning data augmentation approaches for the Kather medical dataset. Once again, the Conditional GAN-based approach, StyleGAN3, efficiently captured the original underlying training distribution, including the parabolic-like curve of ground-truth points located on the right side of the projection area. Random Erasing, while less accurate than StyleGAN3, comes closest to preserving the original projected data. The remaining augmentation approaches, however, struggle to separate clusters effectively, leading to more dispersed distributions throughout the entire visualization area.

Finally, in Figure 8, we present the t-SNE plots for all augmentation approaches applied to our third dataset: the Brazilian Coffee Scenes remote sensing collection. Notice that the conventional deep learning augmentation approaches, such as RandAugment and AutoAugment, were closest to the underlying training distribution. However, as indicated in Table 5, these approaches resulted in the lowest performance gains. Therefore, matching the underlying distribution does not always guarantee improved performance. Conversely, although the distributions of StyleGAN3 and Random Erasing were shifted relative to the original ground-truth distribution, they achieved the highest performance gains when quantitative assessments and multidimensional projections were combined.

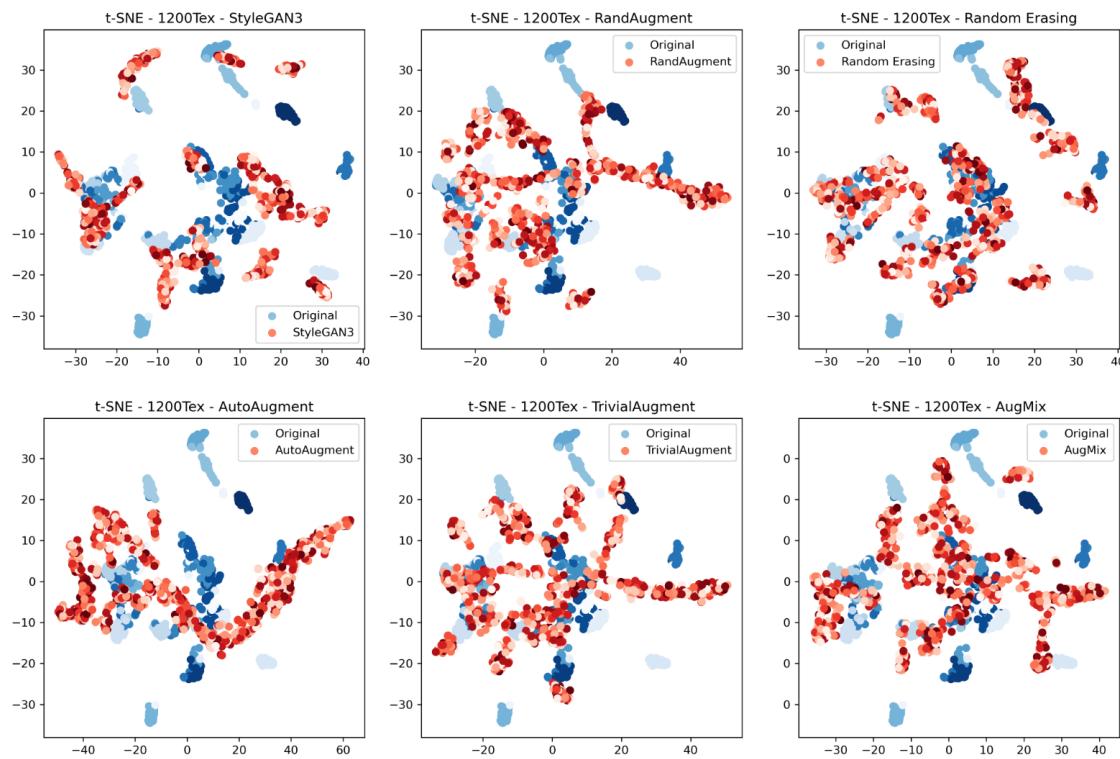


Figure 6. The t-SNE projections of all compared augmentation approaches for the 1200Tex dataset.

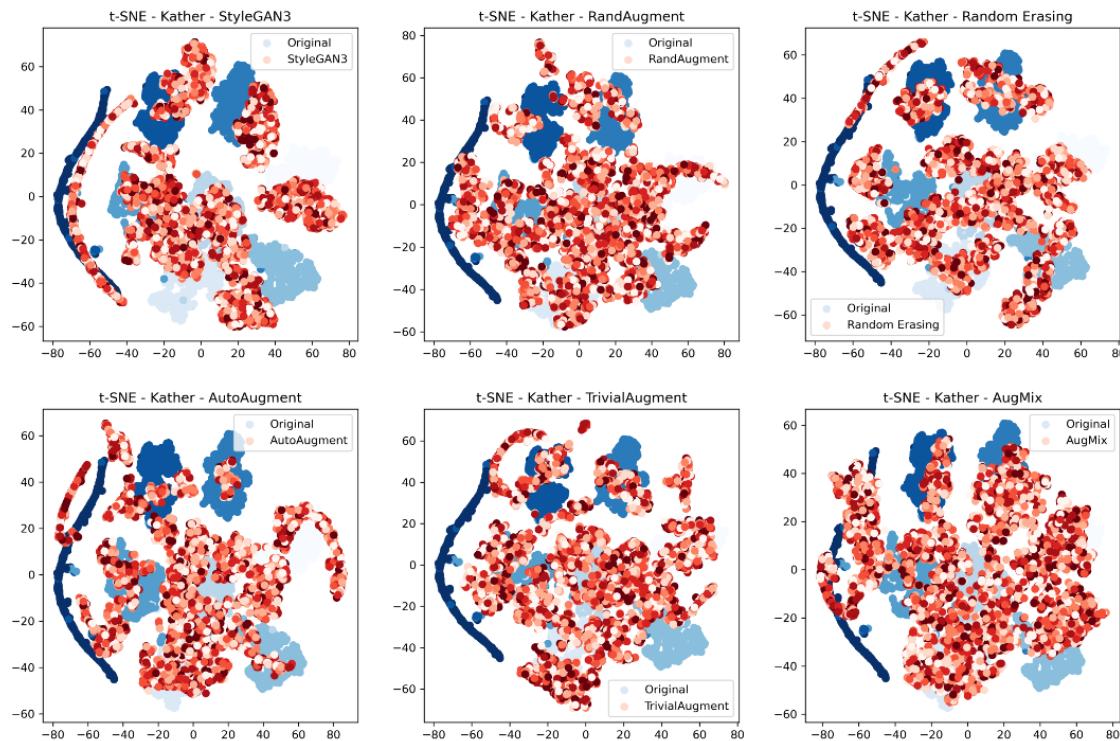


Figure 7. The t-SNE projections of all compared augmentation approaches for the Kather dataset.

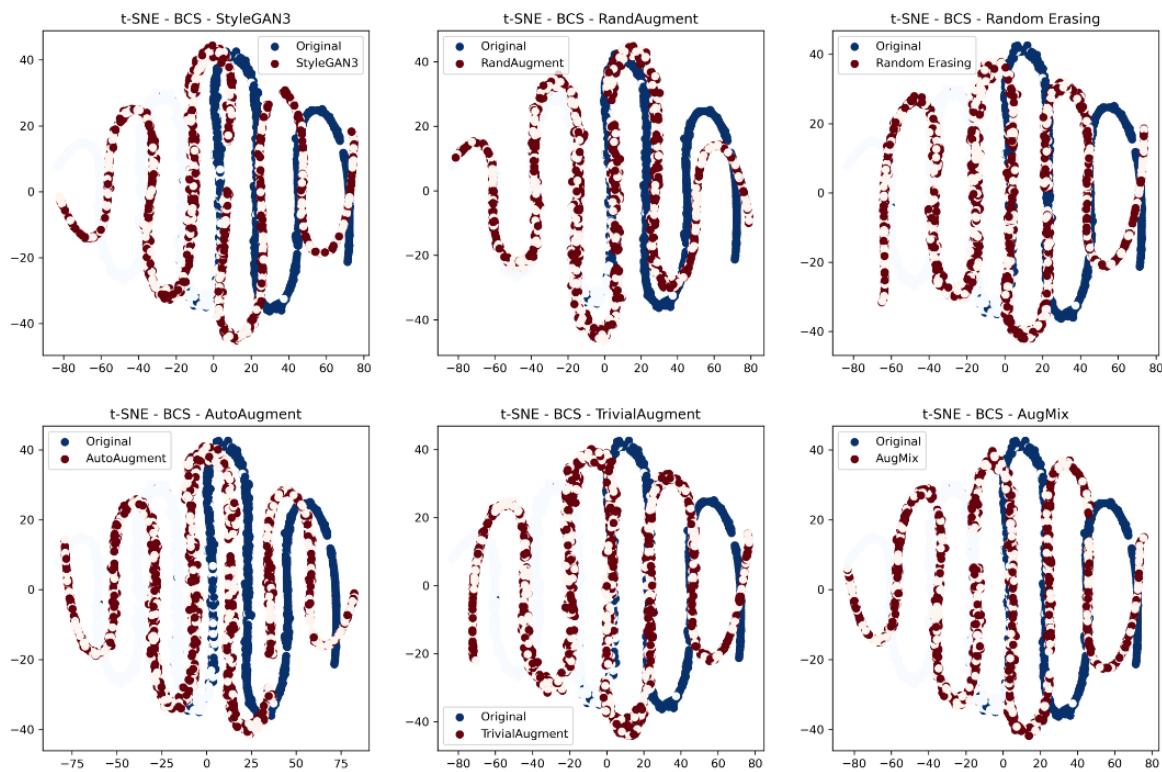


Figure 8. The t-SNE projections of all compared augmentation approaches for the Brazilian Coffee Scenes dataset.

5. Conclusions

In this paper, we investigate the effectiveness of eight deep learning data augmentation techniques in enhancing the generalization capabilities of different classification architectures. Specifically, we assess the robustness of StyleGAN3 [31], DCGAN [32], SAGAN [33], RandAugment [34], Random Erasing [35], AutoAugment [36], TrivialAugment [37] and AugMix [23], while also testing five augmentation factors across four popular deep learning classification networks: two with small depth (ResNet18 [38] and ConvNeXt-Nano [39]), and two with larger depth (DenseNet121 [40] and InceptionResNetV2 [41]). We also compare three representative datasets from different application contexts, including leaf textures, medical images and remote sensing scenes, providing a comprehensive evaluation of all augmentation approaches while emphasizing their benefits and weaknesses.

Our findings demonstrated that StyleGAN3 and Random Erasing were the most effective and stable approaches for all examined datasets and deep learning models, highlighting their potential applicability to other domains of interest. Performance gains of up to 17.23% were achieved by applying the augmentation methods, demonstrating that artificially expanding the datasets is a straightforward yet highly effective strategy for enriching small datasets, particularly in scenarios where data scarcity is a concern. On the other hand, a disadvantage of GAN-based augmentation methods is their higher computational cost during training. However, once the model is trained, generating new data becomes relatively inexpensive and enables the efficient production of an unlimited amount of augmented data. This highlights an important trade-off between computational efficiency and performance that must be carefully considered.

In summary, our battery of experiments showed that deep learning-based data augmentation approaches are effective in handling reduced datasets, highlighting the consistency of StyleGAN3 and Random Erasing across various data collections, with deeper

models consistently outperforming shallower ones in terms of accuracy. Our data-driven analysis offers practical guidance for readers interested in applying deep learning augmentation approaches to small datasets without relying on pre-trained networks, demonstrating the potential of these techniques in data-scarce scenarios.

Finally, future research could focus on how data augmentation techniques can enhance self-supervised learning frameworks by generating diverse and meaningful pretext tasks, thus improving performance in downstream tasks. Another potential direction is to investigate domain-specific applications in fields such as medical imaging and remote sensing, where data scarcity and the need for diversity present unique challenges.

Author Contributions: Conceptualization—L.C.R., W.C. and R.T.F.; investigation—L.C.R., W.C. and R.T.F.; methodology—L.C.R. and R.T.F.; validation—L.C.R. and R.T.F.; resources—L.C.R. and W.C.; funding acquisition—L.C.R. and W.C.; writing—original draft—L.C.R., W.C. and R.T.F. All authors have read and agreed to the published version of the manuscript.

Funding: This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior—Brasil (CAPES); the São Paulo Research Foundation (FAPESP) under grants #2013/07375-0, #2023/14427-8, #2023/04583-2 and #2024/01744-8; and the National Council for Scientific and Technological Development (CNPq), grant #316228/2021-4.

Data Availability Statement: The datasets are available on the Internet. The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

- Agarwal, D.; Berbís, M.Á.; Luna, A.; Lipari, V.; Ballester, J.B.; de la Torre-Díez, I. Automated medical diagnosis of Alzheimer's disease using an efficient net convolutional neural network. *J. Med. Syst.* **2023**, *47*, 57. [[CrossRef](#)] [[PubMed](#)]
- Hasan, M.M.; Hossain, M.M.; Rahman, M.M.; Azad, A.; Alyami, S.A.; Moni, M.A. FP-CNN: Fuzzy pooling-based convolutional neural network for lung ultrasound image classification with explainable AI. *Comput. Biol. Med.* **2023**, *165*, 107407. [[CrossRef](#)] [[PubMed](#)]
- El Alami, A.; Mesbah, A.; Berrahou, N.; Lakhili, Z.; Berrahou, A.; Qjidaa, H. Quaternion discrete orthogonal Hahn moments convolutional neural network for color image classification and face recognition. *Multimed. Tools Appl.* **2023**, *82*, 32827–32853. [[CrossRef](#)]
- Sun, Z.; Feng, C.; Patras, I.; Tzimiropoulos, G. LAFS: Landmark-based Facial Self-supervised Learning for Face Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 1639–1649.
- Xie, L.; Wisse, L.E.; Wang, J.; Ravikumar, S.; Khandelwal, P.; Glenn, T.; Luther, A.; Lim, S.; Wolk, D.A.; Yushkevich, P.A. Deep label fusion: A generalizable hybrid multi-atlas and deep convolutional neural network for medical image segmentation. *Med. Image Anal.* **2023**, *83*, 102683. [[CrossRef](#)] [[PubMed](#)]
- Bruzadin, A.; Boaventura, M.; Colnago, M.; Negri, R.G.; Casaca, W. Learning label diffusion maps for semi-automatic segmentation of lung CT images with COVID-19. *Neurocomputing* **2023**, *522*, 24–38. [[CrossRef](#)]
- Shen, Y.; Liu, D.; Chen, J.; Wang, Z.; Wang, Z.; Zhang, Q. On-board multi-class geospatial object detection based on convolutional neural network for High Resolution Remote Sensing Images. *Remote Sens.* **2023**, *15*, 3963. [[CrossRef](#)]
- Sharma, T.; Chehri, A.; Fofana, I.; Jadhav, S.; Khare, S.; Debaque, B.; Duclos-Hindie, N.; Arya, D. Deep Learning-Based Object Detection and Classification for Autonomous Vehicles in Different Weather Scenarios of Quebec, Canada. *IEEE Access* **2024**, *12*, 13648–13662. [[CrossRef](#)]
- Connor, S.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 1–48. [[CrossRef](#)]
- Lones, M.A. Avoiding common machine learning pitfalls. *Patterns* **2024**, *5*, 101046. [[CrossRef](#)]
- Demirhan, H. A deep learning framework for prediction of crop yield in Australia under the impact of climate change. *Inf. Process. Agric.* **2024**. [[CrossRef](#)]
- Hellín, C.J.; Olmedo, A.A.; Valledor, A.; Gómez, J.; López-Benítez, M.; Tayebi, A. Unraveling the Impact of Class Imbalance on Deep-Learning Models for Medical Image Classification. *Appl. Sci.* **2024**, *14*, 3419. [[CrossRef](#)]

13. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958. [[CrossRef](#)]
14. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on International Conference on Machine Learning, ICML’15, Lille, France, 6–11 July 2015; Volume 37, pp. 448–456.
15. Ioffe, S. Batch Renormalization: Towards Reducing Minibatch Dependence in Batch-Normalized Models. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30, pp. 1945–1953.
16. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 9. [[CrossRef](#)]
17. Shao, L.; Zhu, F.; Li, X. Transfer learning for visual categorization: A survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *26*, 1019–1034. [[CrossRef](#)]
18. Hendrycks, D.; Lee, K.; Mazeika, M. Using pre-training can improve model robustness and uncertainty. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 2712–2721.
19. Zarif, S.; Abdulkader, H.; Elaraby, I.; Alharbi, A.; Elkilani, W.S.; Pławiak, P. Using hybrid pre-trained models for breast cancer detection. *PLoS ONE* **2024**, *19*, e0296912. [[CrossRef](#)]
20. Antoniou, A.; Storkey, A.; Edwards, H. Data augmentation generative adversarial networks. *arXiv* **2017**, arXiv:1711.04340. [[CrossRef](#)]
21. Taylor, L.; Nitschke, G. Improving deep learning with generic data augmentation. In Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI), Bangalore, India, 18–21 November 2018; pp. 1542–1547.
22. Kim, E.K.; Lee, H.; Kim, J.Y.; Kim, S. Data augmentation method by applying color perturbation of inverse PSNR and geometric transformations for object recognition based on deep learning. *Appl. Sci.* **2020**, *10*, 3755. [[CrossRef](#)]
23. Hendrycks, D.; Mu, N.; Cubuk, E.D.; Zoph, B.; Gilmer, J.; Lakshminarayanan, B. AugMix: A Simple Method to Improve Robustness and Uncertainty under Data Shift. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26 April–1 May 2020.
24. Beddiar, D.R.; Oussalah, M.; Muhammad, U.; Seppänen, T. A Deep learning based data augmentation method to improve COVID-19 detection from medical imaging. *Knowl.-Based Syst.* **2023**, *280*, 110985. [[CrossRef](#)]
25. Chao, X.; Sun, G.; Zhao, H.; Li, M.; He, D. Identification of apple tree leaf diseases based on deep learning models. *Symmetry* **2020**, *12*, 1065. [[CrossRef](#)]
26. Montaha, S.; Azam, S.; Rafid, A.; Hasan, M.Z.; Karim, A.; Hasib, K.M.; Patel, S.K.; Jonkman, M.; Mannan, Z.I. MNet-10: A robust shallow convolutional neural network model performing ablation study on medical images assessing the effectiveness of applying optimal data augmentation technique. *Front. Med.* **2022**, *9*, 924979. [[CrossRef](#)]
27. Iglesias, G.; Talavera, E.; Díaz-Álvarez, A. A survey on GANs for computer vision: Recent research, analysis and taxonomy. *Comput. Sci. Rev.* **2023**, *48*, 100553. [[CrossRef](#)]
28. Poojary, R.; Raina, R.; Mondal, A.K. Effect of data-augmentation on fine-tuned CNN model performance. *IAES Int. J. Artif. Intell.* **2021**, *10*, 84. [[CrossRef](#)]
29. Provencher, B.; Badran, A.; Kroll, J.; Marsh, M. Hyperparameter tuning for deep learning semantic image segmentation of micro computed tomography scanned fiber-reinforced composites. *Tomogr. Mater. Struct.* **2024**, *5*, 100032. [[CrossRef](#)]
30. Mera, K.; Sakane, T.; Kurosawa, Y.; Toshiyuki, T. Comparative Analysis of Learning Data Augmentation Techniques for Speech Emotion Recognition. In Proceedings of the Annual Conference of JSAI, Shizuoka, Japan, 28–31 May 2024; Volume JSAI2024, pp. 1–3.
31. Karras, T.; Aittala, M.; Laine, S.; Härkönen, E.; Hellsten, J.; Lehtinen, J.; Aila, T. Alias-free generative adversarial networks. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–14 December 2021; Volume 34, pp. 852–863.
32. Radford, A. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
33. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 7354–7363.
34. Cubuk, E.D.; Zoph, B.; Shlens, J.; Le, Q.V. Randaugment: Practical automated data augmentation with a reduced search space. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 702–703.
35. Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random erasing data augmentation. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 13001–13008.
36. Cubuk, E.D.; Zoph, B.; Mane, D.; Vasudevan, V.; Le, Q.V. Autoaugment: Learning augmentation strategies from data. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 113–123.
37. Müller, S.G.; Hutter, F. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In Proceedings of the IEEE International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 774–782.

38. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NA, USA, 26 June–1 July 2016; pp. 770–778.
39. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A convnet for the 2020s. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 11976–11986.
40. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
41. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
42. Srinivasu, P.N.; SivaSai, J.G.; Ijaz, M.F.; Bhoi, A.K.; Kim, W.; Kang, J.J. Classification of skin disease using deep learning neural networks with MobileNet V2 and LSTM. *Sensors* **2021**, *21*, 2852. [[CrossRef](#)]
43. Zhang, F.; Pan, B.; Shao, P.; Liu, P.; Shen, S.; Yao, P.; Xu, R.X.; Alzheimer’s Disease Neuroimaging Initiative. A single model deep learning approach for Alzheimer’s disease diagnosis. *Neuroscience* **2022**, *491*, 200–214. [[CrossRef](#)]
44. Raj, R.; Mathew, J.; Kannath, S.K.; Rajan, J. Crossover based technique for data augmentation. *Comput. Methods Programs Biomed.* **2022**, *218*, 106716. [[CrossRef](#)]
45. Anwar, T.; Zakir, S. Effect of image augmentation on ECG image classification using deep learning. In Proceedings of the 2021 International Conference on Artificial Intelligence (ICAI), Islamabad, Pakistan, 5–7 April 2021; pp. 182–186.
46. Min, B.; Kim, T.; Shin, D.; Shin, D. Data augmentation method for plant leaf disease recognition. *Appl. Sci.* **2023**, *13*, 1465. [[CrossRef](#)]
47. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232.
48. Branikas, E.; Murray, P.; West, G. A novel data augmentation method for improved visual crack detection using generative adversarial networks. *IEEE Access* **2023**, *11*, 22051–22059. [[CrossRef](#)]
49. Liu, R.; Liu, W.; Zheng, Z.; Wang, L.; Mao, L.; Qiu, Q.; Ling, G. Anomaly-GAN: A data augmentation method for train surface anomaly detection. *Expert Syst. Appl.* **2023**, *228*, 120284. [[CrossRef](#)]
50. Bird, J.J.; Barnes, C.M.; Manso, L.J.; Ekárt, A.; Faria, D.R. Fruit quality and defect image classification with conditional GAN data augmentation. *Sci. Hortic.* **2022**, *293*, 110684. [[CrossRef](#)]
51. Zhou, Q.; Situ, Z.; Teng, S.; Chen, G. Comparative effectiveness of data augmentation using traditional approaches versus styleGANs in automated sewer defect detection. *J. Water Resour. Plan. Manag.* **2023**, *149*, 04023045. [[CrossRef](#)]
52. Su, K.; Zhou, E.; Sun, X.; Wang, C.; Yu, D.; Luo, X. Pre-trained StyleGAN based data augmentation for small sample brain CT motion artifacts detection. In Proceedings of the International Conference on Advanced Data Mining and Applications, Foshan, China, 12–14 November 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 339–346.
53. Nanni, L.; Paci, M.; Brahma, S.; Lumini, A. Comparison of different image data augmentation approaches. *J. Imaging* **2021**, *7*, 254. [[CrossRef](#)]
54. Naveed, H.; Anwar, S.; Hayat, M.; Javed, K.; Mian, A. Survey: Image mixing and deleting for data augmentation. *Eng. Appl. Artif. Intell.* **2024**, *131*, 107791. [[CrossRef](#)]
55. Casanova, D.; de Mesquita Sá Junior, J.J.; Bruno, O.M. Plant leaf identification using Gabor wavelets. *Int. J. Imaging Syst. Technol.* **2009**, *19*, 236–243. [[CrossRef](#)]
56. Kather, J.; Weis, C.; Bianconi, F.; Melchers, S.; Schad, L.; Gaiser, T.; Marx, A.; Zöllner, F. Multi-class texture analysis in colorectal cancer histology. *Sci. Rep.* **2016**, *6*, 27988. [[CrossRef](#)]
57. Penatti, O.A.; Nogueira, K.; Dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, USA, 7–12 June 2015; pp. 44–51.
58. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NA, USA, 26 June–1 July 2016; pp. 2818–2826.
59. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; Volume 27, pp. 1–9.
60. Sharma, P.; Kumar, M.; Sharma, H.K.; Biju, S.M. Generative adversarial networks (GANs): Introduction, Taxonomy, Variants, Limitations, and Applications. *Multimed. Tools Appl.* **2024**, *83*, 88811–88858. [[CrossRef](#)]
61. Gui, J.; Sun, Z.; Wen, Y.; Tao, D.; Ye, J. A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE Trans. Knowl. Data Eng.* **2021**, *35*, 3313–3332.
62. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 4401–4410.

63. Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; Aila, T. Analyzing and improving the image quality of stylegan. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Virtual, 14–19 June 2020; pp. 8110–8119.
64. Karras, T.; Aittala, M.; Laine, S.; Härkönen, E.; Hellsten, J.; Lehtinen, J.; Aila, T. StyleGAN3. Available online: <https://github.com/NVlabs/stylegan3> (accessed on 4 February 2025).
65. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015; p. 15.
66. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30, pp. 6626–6637.
67. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training gans. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; Volume 29, pp. 2226–2234.
68. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
69. Van Der Maaten, L. Learning a parametric embedding by preserving local structure. In Proceedings of the Artificial Intelligence and Statistics, PMLR, Clearwater Beach, FL, USA, 16–18 April 2009; pp. 384–391.
70. Arora, S.; Hu, W.; Kothari, P.K. An analysis of the t-sne algorithm for data visualization. In Proceedings of the Conference on Learning Theory, PMLR, Stockholm, Sweden, 6–9 July 2018; pp. 1455–1462.
71. Campbell, A.; Caudle, K.; Hoover, R.C. Examining intermediate data reduction algorithms for use with t-SNE. In Proceedings of the International Conference on Compute and Data Analysis, Kahului, HI, USA, 14–17 March 2019; pp. 36–42.
72. Nguyen, H.A.T.; Pham, D.H.; Ahn, Y. Effect of Data Augmentation Using Deep Learning on Predictive Models for Geopolymer Compressive Strength. *Appl. Sci.* **2024**, *14*, 3601. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.