DACov: A Deeper Analysis of Data Augmentation on the Computed Tomography Segmentation Problem

Bruno A. Krinski*, Daniel V. Ruiz, Rayson Laroca, Eduardo Todt

Federal University of Paraná, Curitiba, Brazil

ABSTRACT

Due to the COVID-19 global pandemic, computer-assisted diagnoses of medical images have gained much attention, and robust methods of semantic segmentation of Computed Tomography (CT) images have become highly desirable. In this work, we present a deeper analysis of how data augmentation techniques improve segmentation performance on this problem. We evaluate 20 traditional augmentation techniques (i.e., not based on neural networks) on five public datasets. Six different probabilities of applying each augmentation technique on an image were evaluated. We also assess a different training methodology where the training subsets are combined into a single larger set. All networks were evaluated through a 5-fold cross-validation strategy, resulting in over 4,600 experiments. We also propose a novel data augmentation technique based on Generative Adversarial Networks (GANs) to create new healthy and unhealthy lung CT images, evaluating four variations of our approach with the same six probabilities of the traditional methods. Our findings show that GAN-based techniques and spatial-level transformations are the most promising for improving the learning of deep models on this problem, with the StarGAN v2 + F with a probability of 0.3 achieving the highest F-score value on the Ricord1a dataset in the unified training strategy. Our code is publicly available at `https://github.com/VRI-UFPR/DACov2022`.

KEYWORDS

COVID-19, Computed Tomography Segmentation, Data Augmentation

1. Introduction

Since 2019 the world has struggled with the coronavirus pandemic (COVID-19), with millions of infections and deaths worldwide (Wang et al. 2020). According to Johns Hopkins University School of Medicine (2022) (last updated on 30 Dec. 2022), there are more than 659M cases and more than 6.6M deaths globally. Due to the rapid spread of the virus, early diagnosis is highly desirable for faster treatment and screening of infected people (Chen et al. 2020a; Huang et al. 2020).

Automatic detection of COVID-19 infections in Computed Tomography (CT) scans has shown to be a great help for early diagnoses (Shi et al. 2021), with the Semantic Segmentation (Cao and Bao 2020) of CT scans with deep learning-based approaches being widely explored since the COVID-19 outbreak (Shi et al. 2021; Narin et al. 2021). Furthermore, performing automatic segmentation of COVID-19 CT images assists doctors in diagnosing and quantifying COVID-19 lesions in a way to avoid human

---

subjectivity (Zhang et al. 2022; Anthimopoulos et al. 2016). Note that the spreading velocity of the virus increases the number of infected and causes a massive shortage of test kits for Reverse-Transcription Polymerase Chain Reaction (RT-PCR), the primary tool for diagnosing COVID-19. It is also worth mentioning that RT-PCR tests have high false negative rates (Ai et al. 2020). Thus, deep learning methods for COVID-19 CT segmentation have become an essential supplementary tool for RT-PCR tests (Zhang et al. 2022).

However, this process has two main limiting factors. The first one is that labeling images for Semantic Segmentation is a laborious and timing-consuming task, as each image pixel must receive the correct label; to illustrate, for assembling the Cityscapes dataset of urban scenes semantic segmentation (Cordts et al. 2016), an average of 90 minutes was required to label each image. Otherwise, the network may converge to incorrect results (Shi et al. 2021; Cao and Bao 2020). The other limiting factor lies in the fact that the annotation of CT segmentation datasets must be made by highly specialized doctors so that the lesion regions of the image are properly labeled (Shi et al. 2021).

In this work, we propose an extensive analysis of how data augmentation techniques improve the training of Semantic Segmentation networks on this specific problem. A total of 20 traditional different data augmentation techniques and 4 that are Generative Adversarial Network (GAN) based (variations of our method proposed here) applied with six distinct probabilities were evaluated on five different datasets: MedSeg (MedSeg 2021), Zenodo (Jun et al. 2020), CC-CCII (Zhang et al. 2020), MosMed (Morozov et al. 2020), Ricord1a (Tsai et al. 2020), see Section 3.2. A unified training strategy, where all training sets were combined, was also evaluated with the 24 data augmentation techniques (see Section 3.3). All networks were validated through a 5-fold cross-validation strategy, thus resulting in over 4,600 experiments. Additionally, we propose a novel data augmentation technique that exploits a GAN to create new healthy and unhealthy lung CT images (see Section 3.4), and we evaluate 4 variations of our method with the same probabilities of the traditional techniques. Our findings show that GANs based techniques and spatial-level transformations are the most promising for improving the learning of neural networks on this problem, with the StarGAN v2 + F with a probability 0.3 achieving the highest F-score on the Ricord1a dataset in the unified training strategy. The code for running these same experiments is publicly available[1].

A preliminary version of this work was published at the 22th *Simpósio Brasileiro de Computação Aplicada à Saúde* (Brazilian Symposium on Computing Applied to Health) (Krinski et al. 2022). This work differs from that in several aspects. For this work, we used the original CT image resolution of 512×512 instead of 256×256 pixels to reduce the information loss generated from the resize operation, which increased the F-score on four datasets in comparison with results obtained in previous work (Krinski et al. 2022); experiments with six probabilities (0.05, 0.1, 0.15, 0.2, 0.25, 0.3) were conducted and analyzed instead of the previous two (0.1, 0.2); We chose this range to include the probability 0.15 previously used on the literature (Müller et al. 2020, 2021). We improved the stop criteria for training (i.e., instead of relying on a fixed number of epochs, we adopted the early stopping technique); we present and evaluate a different methodology where the training subsets from different datasets are combined into a single larger set, and we propose a novel technique for synthesizing CT images of lungs with and without lesions. Lastly, here we describe and discuss our experiments in a broader and deeper manner.

---

[1]Our code is publicly available at `https://github.com/VRI-UFPR/DACov2022`

2. Related Work

Data augmentation aims to generate synthetic images by applying different operations to preexisting labeled images to aid the learning process of deep learning algorithms in problems that lack available training data (Ruiz et al. 2019, 2020; Zhong et al. 2020; Chen et al. 2020b), with COVID-19 being one of those problems. However, as COVID-19 CT segmentation is a recent problem, few works proposed data augmentation techniques and, in general, these works are limited to generic data augmentation applications (Narin et al. 2021; Diniz et al. 2021; Zhang et al. 2022; Salama and Aly 2022; yan Yao et al. 2022). The most common operation are variations of an affine transformation, such as random flipping, translation, rotation, and scaling. Müller et al. (2020, 2021) evaluated eight generic data augmentation divided into three categories: spatial, color and noise transformations.

Besides generic data augmentation techniques demonstrating promising results, some works prefer to deal with domain adaptation to expand the data available in innovative ways. In (Yazdekhasty et al. 2021), the authors used a conditional GAN (Mirza and Osindero 2014) to generate CT images. First, they extracted the lesion regions from the images, generating images with only the lesions. Then, they mirrored these images and fed them into the GAN model to generate new synthetic images of lesions. Afterward, the generated lesions replaced the original lesions in the original image. The generated images were used to train a Fully Convolutional Network (FCN) architecture for extracting contextual information from the images.

Chen et al. (2023) also proposed a data augmentation strategy for the COVID-19 CT segmentation problem. They employed the Fourier transformation to convert CT scans from cancer patients to CT scans with COVID-19 lesions. Then, a teacher-student architecture was applied to generate segmentation masks for the new COVID-19 CT scans. The student architecture receives the cancer CT scans converted into COVID-19 CT scans, and a set of unlabeled CT scans with COVID-19 lesions and outputs the segmentation masks. While the student network achieved good results for cancer CT segmentation, good results for COVID-19 were not reached. To overcome this problem, the authors added a teacher architecture trained with unlabeled COVID-19 CT scans to help the student architecture extract robust features of COVID-19.

In (Jiang et al. 2021), a data augmentation based on GANs for COVID-19 datasets is proposed. The proposed GAN model is designed with the generator and discriminator as dual networks for global and local feature extraction. Also, the generator contains two sub-discriminators to distinguish multi-resolution images. A dynamic element-wise sum process, called DESUM, was proposed to balance the information extracted in the generator step. A dynamic feature matching process, called DFM, was proposed to weight the loss of input with different resolutions dynamically.

The data augmentation technique proposed by Mahapatra and Singh (2021) uses geometric information of COVID-19 lesions to generate new samples. In their method, the input image X is fed into a Weakly-Supervised Segmentation (WSS) module to generate a segmentation mask S_X. Then, S_X inputs a Spatial Transformer Network (STN) (Jaderberg et al. 2015) to generate a new segmentation mask changing the shape, location, scale, and orientation of the COVID lesions. The output of the STN model inputs a GAN. The generator step of the GAN model uses the STN mask to generate a new sample image which inputs the discriminator. The discriminator is composed of two classifier networks to evaluate the accuracy of the class and shape of the new image.

While there are some explorations in mitigating issues related to the small sizes of

COVID-19 datasets (in terms of the number of images) (Müller et al. 2020, 2021; Zhang et al. 2022), an extensive evaluation of the impact of applying various data augmentation techniques on improving the semantic segmentation performance across multiple COVID-19-related datasets is still lacking. We consider that such an assessment can provide a better insight into the generalization and real improvement of deep networks for this task. Therefore, in this work, we present an extensive benchmark of 20 traditional data augmentation techniques and 4 based on GANs, totalling 24 augmentation techniques. All techniques were applied with six probabilities (0.05, 0.1, 0.15, 0.2, 0.25, and 0.30) on five public datasets, using a 5-fold cross-validation strategy, thus resulting in over 4,600 experiments, which is the largest benchmark on this field to the best of our knowledge.

3. Proposed Work

In this work, we evaluate 20 traditional data augmentation techniques (i.e., not based on neural networks) on COVID-19 CT scans. The techniques were applied to an encoder-decoder network composed of RegNetx-002 (Xu et al. 2022b) and U-net++ (Zhou et al. 2018), comparing six different probabilities of applying the techniques (0.05, 0.1, 0.15, 0.2, 0.25 and 0.30). The input was kept at the original resolution of 512×512 pixels – a higher resolution than previous works such as (Krinski et al. 2022) –; see details in Section 3.2.

In addition to the traditional training approach, where the training and test sets are disjoint subsets from the same dataset, we propose a different methodology: as illustrated in Figure 1 (right), the training subsets are combined into a single larger set while the testing procedure remains essentially the same. To ensure a fair comparison, the original classes were adapted to **background** and **lesion**. For further details on this methodology, see Section 3.3.
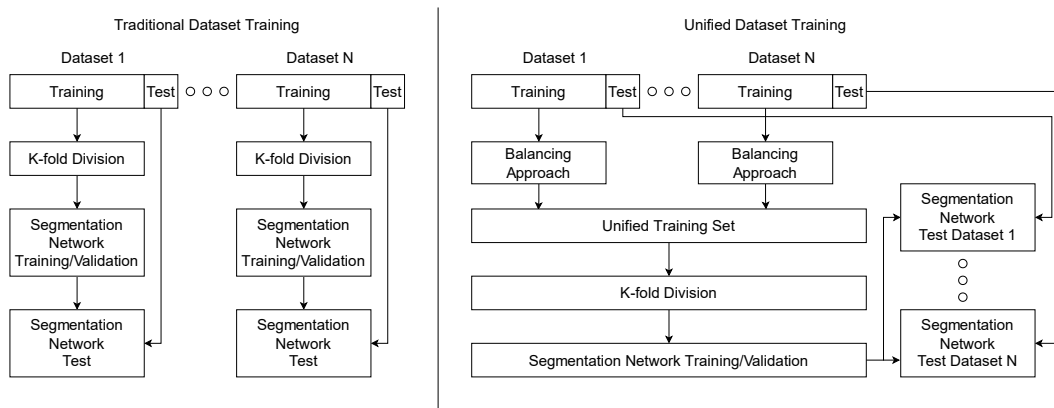

Figure 1. Comparison between the traditional training workflow and our unified training methodology. In the traditional approach (left) the datasets are used to train the network in an individual manner, meanwhile, in the unified training strategy (right) the datasets are combined to train the network.

We also propose a novel data augmentation technique that employs a GAN model to produce new healthy lungs on CT scans (see Figure 2). These new lungs are combined with preexisting labeled lesions to generate new samples and boost the segmentation training; for further details, see Section 3.4. We evaluate four variations of our approach with the same six probabilities of the traditional methods, totaling 24 different techniques evaluated in this paper.
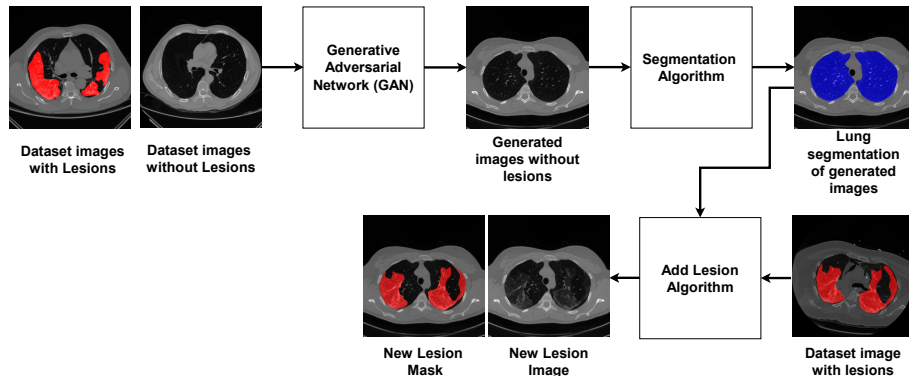
4

Figure 2. Workflow of our approach, which is divided into three main steps. A GAN is trained using two classes, healthy and COVID-19-infected lungs, or only healthy lungs, depending on the network. Thus, the network generates new healthy samples that subsequently have the lung region labeled by a pre-trained segmentation model, ensuring that when the lesions are added later they are cropped by the lung mask. The last step is to add labeled lesions from real CT images into the generated samples. Labeled lesion regions were highlighted in red, labeled lung region was highlighted in blue.

3.1. Datasets and Evaluation Metrics

The segmentation models were trained and evaluated across five different datasets of CT scans: MedSeg (MedSeg 2021), Zenodo (Jun et al. 2020), CC-CCII (Zhang et al. 2020), MosMed (Morozov et al. 2020), and Ricord1a (Tsai et al. 2020). The Ricord1b (Tsai et al. 2020) was used to train the GANs. MedSeg has 929 images and labels for four classes: Background, Ground Glass Opacity (GGO), Consolidation, and Pleural Effusion. The Zenodo dataset has 3,520 images and labels for four classes: Background, Left Lung, Right Lung, and Infections. The MosMed dataset comprises 2,049 images, with labels for two classes: Background and GGO-Consolidation. The Ricord dataset is divided into 1a, 1b, and 1c. Set 1a is the only one with segmentation masks and has 9,166 images with labels for two classes: Background and Infections. Set 1b has 21,220 CT images with negative diagnostics for COVID-19. We also used a subset of CC-CCII with segmentation masks composed of 750 images containing labels for four classes: Background, Lung Field, GGO, and Consolidation. Table 1 summarizes the relevant information about the datasets.

Table 1. The segmentation models were trained and evaluated across five different datasets of CT scans: MedSeg (MedSeg 2021), Zenodo (Jun et al. 2020), CC-CCII (Zhang et al. 2020), MosMed (Morozov et al. 2020) and Ricord1a (Tsai et al. 2020). The Ricord dataset is divided into 1a, 1b, and 1c. Set 1b has images with negative diagnostics for COVID-19 and was used to train the GANs.

Dataset	Type	Number of Images	Removed Images	Labels
CC-CCII	Segmentation	750	201	Background, Lung Field, GGO, and Consolidation
MedSeg	Segmentation	929	457	Background, GGO, Consolidation, and Pleural Effusion
MosMed	Segmentation	2,049	1,264	Background and GGO-Consolidation
Ricord1a	Segmentation	9,166	0	Background and Infections
Ricord1b	Classification	21,220	0	Positive for COVID-19
Zenodo	Segmentation	3,520	1,676	Background, Left Lung, Right Lung, and Infections

One of the problems pointed out in our previous works (Krinski et al. 2021, 2022)

was the class imbalance due to several images with just the background class; in fact, recent work has shown that several applications suffer from class imbalance (Johnson and Khoshgoftaar 2019; Laroca et al. 2021, 2022a; Sanagavarapu et al. 2021). To mitigate this problem, Krinski et al. (2022) removed images with no labeled pixels in the ground-truth mask. However, the CC-CCII and Zenodo datasets remained with masks containing only lung pixels, which still causes an imbalance in the datasets. Therefore, in this work, we removed those images. The total number of removed images in the CC-CCII dataset was 201, and in the Zenodo dataset was 1,676. Also, in the Med-Seg dataset, 457 images were removed; in the MosMed, 1,264 images were removed; and in the Ricord1a, no image was removed. The datasets used in this work do not present predefined subsets for training and testing. Hence, we divided its data into 80% for training and 20% for testing following the Pareto principle (Dunford et al. 2014). This division is also applied to other segmentation problems (Urooj and Borji 2018; Dmitriev and Kaufman 2019; Pandey et al. 2020; Jiwani et al. 2021; Habili et al. 2022), including medical segmentation problems (Dong et al. 2018; Chen et al. 2019; Cao et al. 2020; Li et al. 2022; Gite et al. 2022). In order to perform the training and validation, a standard strategy adopted in the Covid-19 CT segmentation problem is dividing the training set in K-folds, with a 5-fold cross-validation being the most common division adopted (Müller et al. 2021; Saood and Hatem 2021; Yazdekhasty et al. 2021; Fung et al. 2021; Sun et al. 2022). Following that, we use a 5-fold cross-validation strategy avoiding the results being attached to a particular training and validation division. Following (Enshaei et al. 2022; Saeedizadeh et al. 2021), the metrics used for evaluation were the F-score described by Equation (1) and Intersection over Union (IoU) described by Equation (2).

$$F - score = \frac{TruePositive}{TruePositive + \frac{FalsePositive + FalseNegative}{2}} \tag{1}$$

$$IoU = \frac{intersection}{union} \tag{2}$$

The F-score is the harmonic average between Precision and Recall. Precision evaluates the proportion of pixels classified as positive that are genuinely positive, and Recall evaluates the proportion of positive pixels that were indeed classified as positive (Powers 2020). True Positive is the number of positive pixels classified as positive (correct classification of positive pixels); False Positive is the number of negative pixels classified as positive (wrong classification of negative pixels); and False Negative is the number of positive pixels classified as negative (wrong classification of positive pixels).

The IoU (Jaccard 1912) is another metric widely used to evaluate Semantic Segmentation methods (Minaee et al. 2022). The IoU metric takes two areas: the predicted object's area and the target object's area. Then, two values are calculated: these areas' intersection and union. The intersection is the overlap area of the predicted and target objects, and the union is the sum of both areas. Lastly, the IoU is calculated as the intersection value divided by the union value (Minaee et al. 2022).

In order to evaluate the GAN architectures, the Fréchet Inception Distance (FID) (Heusel et al. 2017) was used. This metric compared the distribution of images generated by the GAN models with the distribution of real images used to train the discriminator. FID uses one of the deeper layers of the Inception V3 network

to compare the mean and standard deviation of the distributions. The lower the FID, the better the GAN results.

The one-sided Wilcoxon signed-rank test was applied to perform a statistical analysis of the data augmentation evaluation. The statistical significance is measured through the P-value, which contains the probability of achieving the measured statistical value when the null hypothesis is true. In order to decide to accept or reject the null hypothesis, a significance level $\alpha = 0.05$ is defined (Demšar 2006). Let d represent the difference between the paired samples: $d = x - y$, where x is the distribution without data augmentation, and y is the distribution with data augmentation. Then, if the p-value is higher than α, the null hypothesis is accepted (the underlying distribution d is stochastically higher than a distribution symmetric about zero), or if the P-value is smaller than α, the null hypothesis is rejected.

The estimated computational cost for the entire encoder-decoder network used here is 22.2 Giga Floating Point Operations per Second (GFLOPS) for an input with a resolution of 512×512 pixels. The cost for StyleGAN2 according to Liu et al. (2021); Xu et al. (2022a) is 45.1 GFLOPS for 256×256 with a linear increase for 512×512 pixels. The cost for StarGAN v2 according to Kapoor and Bui (2021) is 120 GFLOPS for 256×256 also with a linear increase for 512×512 pixels. The neural networks used here are implemented using Pytorch (Paszke et al. 2019).

Note that the hardware used for all of our experiments (detailed in Table 2) is a shared resource among researchers in our department.

Table 2. Hardware setup used for all of our experiments.

Machine	Memory	GPU
1	32GB	$1 \times$ NVIDIA TITAN Xp - 12GB
2	512GB	$2 \times$ Tesla P100-SXM2 - 16GB
3	190GB	$4 \times$ Tesla V100-PCIE - 32GB
4	190GB	$4 \times$ Tesla V100-PCIE - 32GB

3.2. Experiment I: Traditional Data Augmentation Techniques

In general, the impacts of Data Augmentation (DA) techniques are not approached in studies proposed for the COVID-19 CT segmentation problem (Saood and Hatem 2021; Mahmud et al. 2021). While some studies apply some data augmentation techniques (Zhao et al. 2021; Qiblawey et al. 2021; Raj et al. 2021; Müller et al. 2020, 2021; Chen et al. 2020c; Xu et al. 2020), most of them are limited to flip and rotation transformations. To correctly measure the impact of data augmentation on the COVID-19 CT segmentation problem, we evaluate 20 different data augmentation techniques not based on neural networks: CLAHE, Coarse Dropout, Elastic Transform, Emboss, Flip, Gaussian Blur, Grid Distortion, Grid Dropout, Image Compression, Median Blur, Optical Distortion, Piecewise Affine, Posterize, RBC, Random Crop, Random Gamma, Random Snow, Rotate, Sharpen, Shift Scale Rotate. Figure 3 illustrates the 20 data augmentation techniques applied to a CT image. These data augmentation techniques were applied – with the default parameters – using the Albumentations library (Buslaev et al. 2020), which has been successfully explored in various areas of computer vision (Kupyn et al. 2019; Kaissis et al. 2021; Laroca et al. 2022b). Parameter optimization of the best techniques was left for future work.

The encoder-decoder network chosen to evaluate the augmentation techniques was
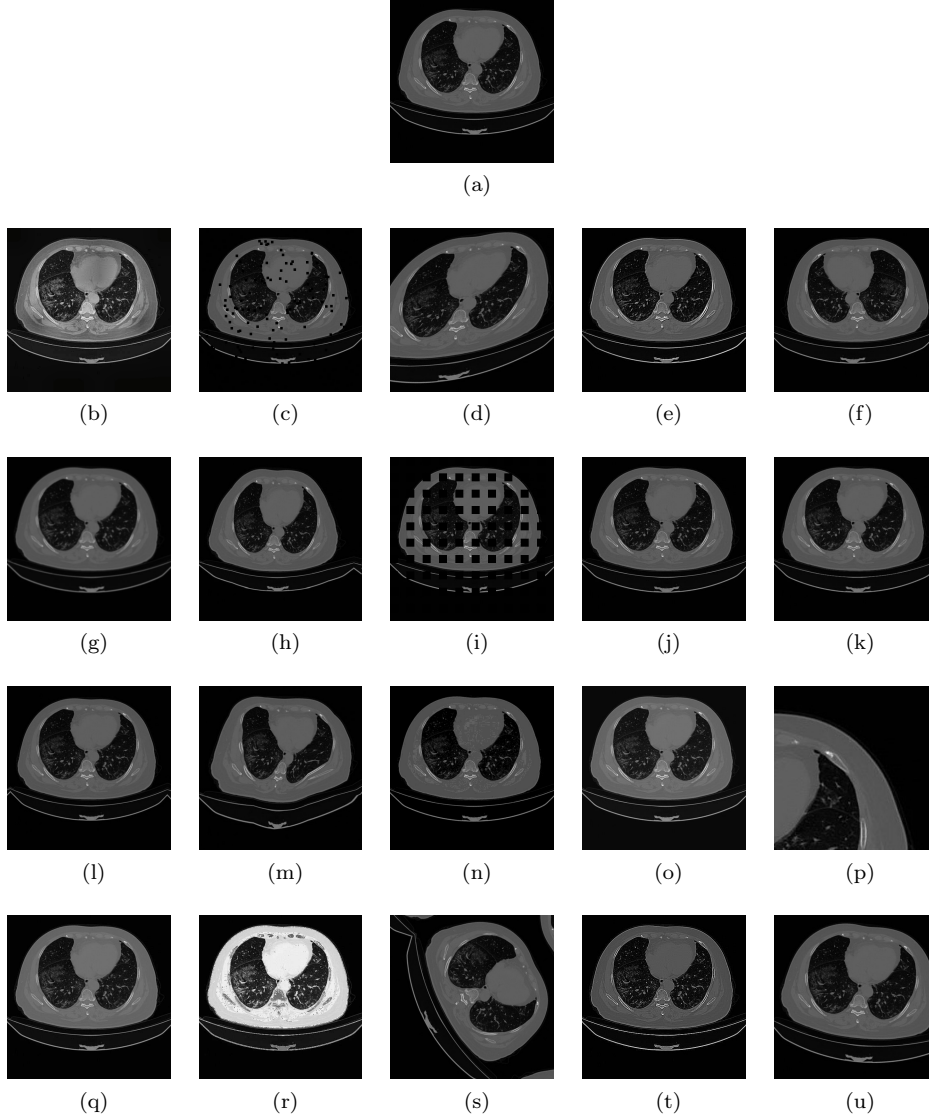
Figure 3. Illustration of the 20 traditional data augmentation techniques (i.e., not based on neural networks) applied to a CT image. In 3a, the original image is presented. The data augmentation techniques are illustrated in the following sequence: CLAHE 3b, Coarse Dropout 3c, Elastic Transform 3d, Emboss 3e, Flip 3f, Gaussian Blur 3g, Grid Distortion 3h, Grid Dropout 3i, Image Compression 3j, Median Blur 3k, Optical Distortion 3l, Piecewise Affine 3m, Posterize 3n, RBC 3o, Random Crop 3p, Random Gamma 3q, Random Snow 3r, Rotate 3s, Sharpen 3t, and Shift Scale Rotate 3u.

the RegNetx-002 (Xu et al. 2022b) encoder and U-net++ (Zhou et al. 2018) decoder. All experiments were evaluated through a 5-fold cross-validation strategy. We trained the chosen architecture for a maximum of 100 epochs, with patience = 10 (the number of epochs with no improvement after which training is stopped). The initial learning rate was 0.001 and was divided by 10 every 10 epochs. The augmentation algorithms were applied through a technique called online augmentation. In this technique, the data augmentation is applied during the network training, and each batch of images has a probability of suffering an augmentation operation before inputting the network. Differently from our previous work (Krinski et al. 2022), here we also analyze the effects of increasing the number of images augmented, with six probabilities of applying the

Table 3. Results of the data augmentation evaluation. p stands for probability, the blue-colored values indicate the best F-score values, while the red-colored values indicate the best IoU values. The values highlighted in green show the data augmentation techniques in which the P-value achieved values lower than 0.05, and thus the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation).

p	Augmentation	CC-CCII		MedSeg		MosMed		Ricord1a		Zenodo	
		F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU
	No Augmentation	0.8450	0.7896	0.8885	0.8261	0.8180	0.7528	0.8875	0.8275	0.9072	0.8490
	CLAHE	0.8364	0.7810	0.8839	0.8211	0.8137	0.7478	0.8871	0.8274	0.9072	0.8488
	Coarse Dropout	0.8463	0.7911	0.8869	0.8247	0.8160	0.7499	0.8872	0.8271	0.9077	0.8498
	Elastic Transform	0.8467	0.7915	0.8886	0.8266	0.8136	0.7482	0.8856	0.8256	0.9073	0.8489
	Emboss	0.8477	0.7916	0.8867	0.8244	0.8162	0.7505	0.8866	0.8268	0.9082	0.8500
	Flip	0.8458	0.7905	0.8869	0.8246	0.8189	0.7542	0.8884	0.8281	0.9087	0.8509
	Gaussian Blur	0.8451	0.7894	0.8856	0.8229	0.8126	0.7467	0.8883	0.8280	0.9086	0.8502
	Grid Distortion	0.8457	0.7897	0.8868	0.8242	0.8197	0.7536	0.8855	0.8252	0.9077	0.8494
	Grid Dropout	0.8391	0.7840	0.8870	0.8245	0.8149	0.7499	0.8866	0.8265	0.9069	0.8486
	Image Compression	0.8386	0.7835	0.8868	0.8248	0.8118	0.7465	0.8881	0.8282	0.9074	0.8491
0.05	Median Blur	0.8446	0.7887	0.8849	0.8225	0.8116	0.7455	0.8888	0.8289	0.9078	0.8492
	Optical Distortion	0.8431	0.7884	0.8865	0.8246	0.8134	0.7485	0.8852	0.8250	0.9081	0.8499
	Piecewise Affine	0.8468	0.7914	0.8874	0.8257	0.8155	0.7498	0.8879	0.8279	0.9076	0.8495
	Posterize	0.8455	0.7899	0.8873	0.8252	0.8168	0.7504	0.8895	0.8299	0.9076	0.8489
	RBC	0.8441	0.7883	0.8872	0.8247	0.8129	0.7474	0.8870	0.8268	0.9084	0.8502
	Random Crop	0.8440	0.7886	0.8853	0.8224	0.8156	0.7500	0.8862	0.8265	0.9067	0.8486
	Random Gamma	0.8431	0.7890	0.8866	0.8244	0.8176	0.7523	0.8889	0.8289	0.9083	0.8499
	Random Snow	0.8432	0.7870	0.8861	0.8237	0.8152	0.7497	0.8863	0.8260	0.9063	0.8476
	Rotate	0.8479	0.7917	0.8885	0.8262	0.8175	0.7523	0.8861	0.8258	0.9084	0.8500
	Sharpen	0.8481	0.7931	0.8873	0.8249	0.8154	0.7504	0.8882	0.8283	0.9085	0.8502
	Shift Scale Rotate	0.8459	0.7894	0.8897	0.8281	0.8185	0.7526	0.8860	0.8259	0.9083	0.8500
	CLAHE	0.8372	0.7816	0.8864	0.8240	0.8125	0.7472	0.8874	0.8273	0.9070	0.8486
	Coarse Dropout	0.8477	0.7922	0.8878	0.8259	0.8171	0.7516	0.8874	0.8272	0.9080	0.8499
	Elastic Transform	0.8481	0.7931	0.8875	0.8255	0.8158	0.7503	0.8871	0.8271	0.9088	0.8510
	Emboss	0.8436	0.7872	0.8880	0.8260	0.8158	0.7503	0.8885	0.8283	0.9088	0.8510
	Flip	0.8483	0.7928	0.8877	0.8260	0.8204	0.7547	0.8861	0.8260	0.9081	0.8495
	Gaussian Blur	0.8436	0.7891	0.8843	0.8213	0.8136	0.7480	0.8874	0.8273	0.9071	0.8487
	Grid Distortion	0.8458	0.7910	0.8889	0.8266	0.8211	0.7564	0.8864	0.8267	0.9088	0.8511
	Grid Dropout	0.8422	0.7869	0.8855	0.8229	0.8152	0.7498	0.8840	0.8237	0.9073	0.8484
	Image Compression	0.8490	0.7938	0.8875	0.8255	0.8150	0.7489	0.8869	0.8268	0.9086	0.8504
0.1	Median Blur	0.8460	0.7907	0.8861	0.8235	0.8167	0.7501	0.8888	0.8288	0.9074	0.8491
	Optical Distortion	0.8483	0.7940	0.8893	0.8275	0.8164	0.7500	0.8888	0.8290	0.9080	0.8498
	Piecewise Affine	0.8505	0.7949	0.8892	0.8276	0.8194	0.7536	0.8871	0.8270	0.9072	0.8488
	Posterize	0.8448	0.7904	0.8862	0.8239	0.8092	0.7435	0.8866	0.8268	0.9077	0.8497
	RBC	0.8415	0.7870	0.8873	0.8249	0.8095	0.7426	0.8871	0.8270	0.9079	0.8495
	Random Crop	0.8375	0.7819	0.8869	0.8248	0.8137	0.7480	0.8877	0.8279	0.9080	0.8499
	Random Gamma	0.8447	0.7901	0.8870	0.8251	0.8118	0.7451	0.8867	0.8268	0.9071	0.8486
	Random Snow	0.8460	0.7906	0.8851	0.8222	0.8114	0.7457	0.8861	0.8258	0.9068	0.8479
	Rotate	0.8503	0.7947	0.8874	0.8256	0.8185	0.7526	0.8848	0.8248	0.9077	0.8491
	Sharpen	0.8492	0.7931	0.8849	0.8224	0.8173	0.7519	0.8890	0.8291	0.9076	0.8494
	Shift Scale Rotate	0.8491	0.7944	0.8905	0.8287	0.8186	0.7537	0.8861	0.8260	0.9086	0.8504

data augmentation being evaluated: 0.05, 0.1, 0.15, 0.2, 0.25, 0.30.

In order to avoid information loss when the image is downscaled, the segmentation network was trained with the original resolution of 512×512 instead of the downscaled 256×256 resolution used in our previous work (Krinski et al. 2022). Downscaling the image generates a loss of information in the images, which negatively affects network learning. Compared with (Krinski et al. 2022), training without downscaling generated impressive gains in F-score in four of the five evaluated datasets. The most significant difference was in the MedSeg dataset with the Shift Scale Rotate augmentation (the best results achieved), where the results increased from 0.4806 with an image size of 256×256 to 0.8890 with an image size of 512×512, a gain of 0.4084 in the F-score value.

Table 3 presents the evaluation results for probabilities 0.05 and 0.1, Table 4 presents the results for probabilities 0.15 and 0.2, and Table 5 presents the results for probabilities 0.25 and 0.3. The values highlighted in green show the data augmentation techniques where the P-value achieved values lower than 0.05, and the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation).

Most data augmentation techniques did not improve the F-score and the IoU val-

Table 4. Results of the data augmentation evaluation (Continuation of Table 3). p stands for probability, the blue-colored values indicate the best F-score values, while the red-colored values indicate the best IoU values. The values highlighted in green show the data augmentation techniques where the P-value achieved values lower than 0.05, and thus the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation).

p	Augmentation	CC-CCII		MedSeg		MosMed		Ricord1a		Zenodo	
		F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU
	No Augmentation	0.8450	0.7896	0.8885	0.8261	0.8180	0.7528	0.8875	0.8275	0.9072	0.8490
0.15	CLAHE	0.8414	0.7863	0.8836	0.8208	0.8117	0.7451	0.8886	0.8288	0.9082	0.8497
	Coarse Dropout	0.8415	0.7869	0.8856	0.8232	0.8118	0.7466	0.8896	0.8297	0.9065	0.8481
	Elastic Transform	0.8477	0.7926	0.8891	0.8273	0.8195	0.7538	0.8857	0.8255	0.9091	0.8512
	Emboss	0.8457	0.7905	0.8880	0.8255	0.8155	0.7504	0.8889	0.8295	0.9080	0.8498
	Flip	0.8478	0.7919	0.8881	0.8255	0.8220	0.7562	0.8846	0.8240	0.9080	0.8489
	Gaussian Blur	0.8462	0.7912	0.8871	0.8246	0.8137	0.7484	0.8883	0.8280	0.9073	0.8491
	Grid Distortion	0.8484	0.7935	0.8882	0.8258	0.8205	0.7541	0.8831	0.8224	0.9083	0.8504
	Grid Dropout	0.8379	0.7819	0.8838	0.8207	0.8152	0.7491	0.8866	0.8263	0.9078	0.8491
	Image Compression	0.8437	0.7879	0.8867	0.8248	0.8148	0.7498	0.8890	0.8294	0.9065	0.8480
	Median Blur	0.8445	0.7898	0.8827	0.8196	0.8141	0.7479	0.8870	0.8273	0.9081	0.8497
	Optical Distortion	0.8506	0.7944	0.8889	0.8273	0.8148	0.7499	0.8863	0.8263	0.9092	0.8513
	Piecewise Affine	0.8485	0.7943	0.8869	0.8248	0.8189	0.7541	0.8861	0.8260	0.9088	0.8509
	Posterize	0.8426	0.7873	0.8862	0.8238	0.8157	0.7498	0.8880	0.8276	0.9071	0.8488
	RBC	0.8449	0.7892	0.8866	0.8240	0.8132	0.7466	0.8860	0.8256	0.9071	0.8485
	Random Crop	0.8447	0.7903	0.8877	0.8254	0.8139	0.7484	0.8872	0.8272	0.9081	0.8498
	Random Gamma	0.8455	0.7909	0.8866	0.8244	0.8134	0.7478	0.8880	0.8277	0.9081	0.8500
	Random Snow	0.8430	0.7861	0.8845	0.8217	0.8170	0.7506	0.8842	0.8240	0.9062	0.8472
	Rotate	0.8491	0.7924	0.8865	0.8239	0.8190	0.7530	0.8842	0.8233	0.9084	0.8500
	Sharpen	0.8472	0.7925	0.8877	0.8255	0.8136	0.7479	0.8865	0.8265	0.9071	0.8490
	Shift Scale Rotate	0.8461	0.7919	0.8888	0.8266	0.8198	0.7542	0.8841	0.8236	0.9083	0.8498
0.2	CLAHE	0.8421	0.7868	0.8847	0.8213	0.8141	0.7480	0.8860	0.8255	0.9072	0.8483
	Coarse Dropout	0.8448	0.7890	0.8857	0.8234	0.8114	0.7465	0.8870	0.8269	0.9081	0.8500
	Elastic Transform	0.8488	0.7932	0.8870	0.8249	0.8198	0.7546	0.8846	0.8242	0.9100	0.8522
	Emboss	0.8516	0.7957	0.8886	0.8266	0.8134	0.7473	0.8873	0.8273	0.9075	0.8491
	Flip	0.8551	0.7988	0.8870	0.8241	0.8231	0.7574	0.8803	0.8192	0.9094	0.8505
	Gaussian Blur	0.8415	0.7872	0.8859	0.8236	0.8131	0.7468	0.8877	0.8273	0.9078	0.8497
	Grid Distortion	0.8533	0.7980	0.8887	0.8268	0.8237	0.7579	0.8862	0.8263	0.9082	0.8499
	Grid Dropout	0.8424	0.7878	0.8849	0.8220	0.8122	0.7458	0.8855	0.8254	0.9072	0.8484
	Image Compression	0.8416	0.7851	0.8864	0.8241	0.8108	0.7462	0.8879	0.8282	0.9071	0.8489
	Median Blur	0.8452	0.7897	0.8863	0.8238	0.8145	0.7483	0.8866	0.8259	0.9067	0.8481
	Optical Distortion	0.8494	0.7937	0.8895	0.8279	0.8163	0.7509	0.8881	0.8285	0.9082	0.8502
	Piecewise Affine	0.8510	0.7959	0.8857	0.8231	0.8233	0.7568	0.8865	0.8263	0.9087	0.8503
	Posterize	0.8478	0.7921	0.8874	0.8250	0.8148	0.7494	0.8871	0.8268	0.9089	0.8508
	RBC	0.8410	0.7858	0.8868	0.8239	0.8110	0.7437	0.8867	0.8263	0.9065	0.8481
	Random Crop	0.8468	0.7909	0.8861	0.8241	0.8164	0.7509	0.8866	0.8269	0.9073	0.8491
	Random Gamma	0.8502	0.7939	0.8861	0.8242	0.8141	0.7483	0.8892	0.8290	0.9083	0.8500
	Random Snow	0.8455	0.7896	0.8814	0.8175	0.8119	0.7456	0.8856	0.8251	0.9056	0.8467
	Rotate	0.8498	0.7941	0.8885	0.8261	0.8238	0.7577	0.8816	0.8203	0.9073	0.8500
	Sharpen	0.8457	0.7900	0.8886	0.8261	0.8131	0.7478	0.8874	0.8273	0.9078	0.8499
	Shift Scale Rotate	0.8512	0.7955	0.8890	0.8271	0.8233	0.7571	0.8830	0.8224	0.9090	0.8507

ues. For the probability of 0.05, no method achieved an F-score improvement higher than 1% compared to the baseline, where no data augmentation technique was applied. However, seven techniques achieved statistical difference in the Ricord1a, and one technique achieved statistical differences in CC-CCII and Zenodo. For probability 0.1, Piecewise Affine and Rotate improved the F-score in the CC-CCII; Shift Scale Rotate improved the F-score on the MedSeg, and Grid Distortion improved the F-score on the MosMed. With a probability of 0.15, Optical Distortion improved the F-score on the CC-CCII. With a probability of 0.2, 6 data augmentation techniques improved the F-score on the CC-CCII, and 5 techniques improved the F-score on the MosMed. Elastic Transform improved the F-score on the Zenodo dataset. With a probability of 0.25, 3 data augmentation techniques reached better F-score values on CC-CCII, and 3 techniques achieved better F-score values on MosMed. With a probability of 0.3, 7 data augmentation techniques improved the F-score value obtained on CC-CCII, 5 techniques improved the F-score achieved on MosMed, and 1 technique (Gaussian Blur) improved the F-score reached on Ricord1a. In general, the probability of 0.3 produced better results considering the number of techniques that improved the F-score values.

In summary, the best result per dataset was: CC-CCII with Shift Scale Rotate

Table 5. Results of the data augmentation evaluation (Continuation of Table 3 and Table 4). p stands for probability, the blue-colored values indicate the best F-score values, while the red-colored values indicate the best IoU values. The values highlighted in green show the data augmentation techniques in which the P-value achieved values lower than 0.05, and thus the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation).

p	Augmentation	CC-CCII		MedSeg		MosMed		Ricord1a		Zenodo	
		F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU
	No Augmentation	0.8450	0.7896	0.8885	0.8261	0.8180	0.7528	0.8875	0.8275	0.9072	0.8490
0.25	CLAHE	0.8396	0.7842	0.8842	0.8211	0.8108	0.7450	0.8869	0.8263	0.9058	0.8469
	Coarse Dropout	0.8406	0.7852	0.8847	0.8219	0.8152	0.7501	0.8858	0.8257	0.9076	0.8494
	Elastic Transform	0.8537	0.7980	0.8887	0.8269	0.8195	0.7534	0.8844	0.8239	0.9096	0.8519
	Emboss	0.8448	0.7892	0.8883	0.8260	0.8169	0.7511	0.8881	0.8279	0.9072	0.8491
	Flip	0.8526	0.7963	0.8851	0.8226	0.8284	0.7631	0.8827	0.8221	0.9082	0.8494
	Gaussian Blur	0.8483	0.7932	0.8852	0.8225	0.8125	0.7465	0.8887	0.8287	0.9073	0.8489
	Grid Distortion	0.8525	0.7971	0.8870	0.8248	0.8227	0.7568	0.8871	0.8269	0.9085	0.8503
	Grid Dropout	0.8398	0.7849	0.8820	0.8189	0.8136	0.7470	0.8837	0.8227	0.9052	0.8456
	Image Compression	0.8396	0.7845	0.8872	0.8253	0.8124	0.7467	0.8857	0.8259	0.9075	0.8494
	Median Blur	0.8439	0.7892	0.8846	0.8218	0.8177	0.7514	0.8834	0.8226	0.9069	0.8490
	Optical Distortion	0.8481	0.7928	0.8883	0.8263	0.8208	0.7547	0.8864	0.8263	0.9089	0.8509
	Piecewise Affine	0.8478	0.7921	0.8868	0.8248	0.8240	0.7581	0.8872	0.8268	0.9087	0.8503
	Posterize	0.8447	0.7891	0.8875	0.8255	0.8183	0.7533	0.8851	0.8245	0.9080	0.8497
	RBC	0.8405	0.7847	0.8863	0.8234	0.8122	0.7448	0.8843	0.8231	0.9064	0.8476
	Random Crop	0.8467	0.7910	0.8870	0.8249	0.8109	0.7454	0.8880	0.8287	0.9070	0.8486
	Random Gamma	0.8392	0.7849	0.8859	0.8232	0.8142	0.7481	0.8884	0.8283	0.9065	0.8483
	Random Snow	0.8439	0.7867	0.8802	0.8160	0.8157	0.7488	0.8860	0.8252	0.9037	0.8443
	Rotate	0.8495	0.7932	0.8879	0.8256	0.8222	0.7561	0.8787	0.8175	0.9071	0.8482
	Sharpen	0.8418	0.7870	0.8891	0.8268	0.8109	0.7454	0.8895	0.8297	0.9080	0.8496
	Shift Scale Rotate	0.8493	0.7951	0.8891	0.8274	0.8188	0.7527	0.8823	0.8214	0.9083	0.8497
0.3	CLAHE	0.8400	0.7846	0.8846	0.8215	0.8085	0.7428	0.8824	0.8215	0.9062	0.8474
	Coarse Dropout	0.8413	0.7860	0.8856	0.8235	0.8114	0.7457	0.8883	0.8283	0.9081	0.8497
	Elastic Transform	0.8546	0.7996	0.8893	0.8273	0.8250	0.7590	0.8847	0.8241	0.9097	0.8520
	Emboss	0.8458	0.7904	0.8881	0.8259	0.8138	0.7480	0.8877	0.8276	0.9078	0.8498
	Flip	0.8490	0.7918	0.8846	0.8213	0.8257	0.7605	0.8792	0.8183	0.9087	0.8496
	Gaussian Blur	0.8442	0.7892	0.8865	0.8243	0.8097	0.7440	0.8903	0.8303	0.9058	0.8471
	Grid Distortion	0.8556	0.7993	0.8894	0.8274	0.8258	0.7593	0.8867	0.8267	0.9089	0.8507
	Grid Dropout	0.8408	0.7854	0.8838	0.8210	0.8131	0.7472	0.8840	0.8231	0.9061	0.8469
	Image Compression	0.8508	0.7945	0.8871	0.8252	0.8138	0.7483	0.8867	0.8269	0.9080	0.8500
	Median Blur	0.8452	0.7907	0.8836	0.8209	0.8143	0.7478	0.8872	0.8272	0.9086	0.8505
	Optical Distortion	0.8527	0.7966	0.8896	0.8277	0.8184	0.7532	0.8890	0.8293	0.9091	0.8513
	Piecewise Affine	0.8561	0.8006	0.8884	0.8266	0.8224	0.7562	0.8846	0.8243	0.9087	0.8505
	Posterize	0.8440	0.7877	0.8865	0.8241	0.8127	0.7472	0.8858	0.8257	0.9065	0.8477
	RBC	0.8421	0.7862	0.8865	0.8239	0.8097	0.7432	0.8866	0.8260	0.9066	0.8478
	Random Crop	0.8476	0.7920	0.8869	0.8248	0.8144	0.7487	0.8867	0.8267	0.9073	0.8490
	Random Gamma	0.8513	0.7949	0.8865	0.8243	0.8128	0.7465	0.8867	0.8267	0.9080	0.8498
	Random Snow	0.8468	0.7901	0.8815	0.8169	0.8148	0.7484	0.8848	0.8237	0.9043	0.8450
	Rotate	0.8494	0.7933	0.8876	0.8248	0.8253	0.7592	0.8789	0.8174	0.9066	0.8472
	Sharpen	0.8423	0.7891	0.8862	0.8231	0.8093	0.7440	0.8866	0.8260	0.9089	0.8512
	Shift Scale Rotate	0.8567	0.8013	0.8887	0.8267	0.8237	0.7571	0.8808	0.8199	0.9092	0.8506

applied with a probability of 0.3, achieving an F-score of 0.8567; MedSeg with Shift Scale Rotate applied with a probability of 0.1, reaching an F-score of 0.8905; MosMed with Flip applied with a probability of 0.25, obtaining an F-score of 0.8284; Ricord1a with Gaussian Blur applied with a probability of 0.3, achieving an F-score of 0.8903; and Zenodo with Elastic Transform applied with a probability of 0.2, attaining an F-score of 0.91. Thus, in the CC-CCII, MedSeg, MosMed and Zenodo datasets the highest F-score was achieved through a spatial transformation, whereas in Ricord1a it was reached with a color operation.

In four datasets, the best augmentation techniques are spatial-based operations. Such techniques provided better results in those datasets because they encourage shape variation of the lesion regions. Unlike other datasets, Ricord1a has very similar images, with minimal changes in lung position and shape. Therefore, applying a spatial operation in this dataset is counterproductive. As a result, a color operation (Gaussian Blur) achieved the highest F-score, indicating that this dataset is more sensitive to color operations.

3.3. Proposed Methodology: Training Sets Unified

In addition to the traditional approach, where the training and test sets are disjoint subsets from the same dataset, we employ a different methodology that combines the training subset from each dataset into a single larger set (See Figure 1). The original classes of the five datasets were rearranged into two classes: **background** and **lesion**, where everything that was not a type of lesion was converted to the background, and all lesions' sub-types were merged into a single class. The reasoning is to encourage generalization since the trained model is expected to perform reasonably well with any CT image of the lungs.

Besides the rearrangement of the classes, an additional balancing procedure was employed to ensure a fair representation of each original sample on the new combined training set. In summary, smaller datasets (in terms of the number of images), i.e., CC-CCII, MedSeg, MosMed, and Zenodo, had their samples repeated n times to match the number of samples of the largest dataset Ricord1a. For each dataset i, n_i was calculated as the ceil of the number of the images in the Ricord1a dataset divided by the number of images in the corresponding dataset x_i, as described by Equation (3).

$$n_i = \left\lceil \frac{|Ricord1a|}{|x_i|} \right\rceil \tag{3}$$

3.3.1. Unified Set Evaluation

Table 6 presents the evaluation results for probabilities 0.05 and 0.1, Table 7 presents the results for probabilities 0.15 and 0.2, and Table 8 presents the results for probabilities 0.25 and 0.3. The values highlighted in green show the data augmentations where the P-value achieved values lower than 0.05, and the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation). Training with a unified training set was also compared with training with individual training sets. The underscored values presented in Table 6, Table 7 and Table 8 show the techniques in which training with the unified set achieved a P-value lower than 0.05, and the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved in the unified training set are better than the individual training sets).

As presented in Table 6, Table 7 and Table 8, using the unified training set achieved promising results compared to using the individual training sets. For a probability of 0.05, training with the unified training set achieved a higher F-score when compared with the baseline in all data augmentation techniques applied in the CC-CCII dataset. The same occurred in 11 data augmentation techniques in MosMed and 10 data in Zenodo.

For a probability of 0.1, the unified training set achieved a better F-score value in 2 data augmentation techniques in the MedSeg dataset, 13 in MosMed, 11 in Zenodo, and all data augmentation techniques in the CC-CCII dataset. For a probability of 0.15, the unified training set achieved a better F-score value in 6 data augmentation techniques in the MedSeg dataset, 17 in MosMed, 13 in Zenodo, and 19 in CC-CCII. For a probability of 0.2, the unified training set achieved better F-score values in 8 data augmentation techniques in MedSeg, 12 in MosMed, 11 in Zenodo, and all data augmentation techniques in the CC-CCII dataset.

For a probability of 0.25, the unified training set achieved a better F-score value in 9 data augmentation techniques in the MedSeg dataset, 17 in MosMed, 15 in Zenodo.

Table 6. Results of the data augmentation evaluation when unifying the training sets. p stands for probability, the blue-colored values indicate the best F-score value, and the red-colored values indicate the best IoU values. The values highlighted in green show the data augmentation techniques in which the P-value achieved values lower than 0.05, and thus the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation). The underscored values show the techniques where training with the unified set achieved a P-value lower than 0.05 when compared with training with a single training set, and the null hypothesis was rejected.

p	Augmentation	CC-CCII		MedSeg		MosMed		Ricord1a		Zenodo	
		F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU
	No Augmentation	0.8636	0.8087	0.8881	0.8253	0.8185	0.7547	0.8599	0.7947	0.9096	0.8514
0.05	CLAHE	0.8580	0.8025	0.8873	0.8247	0.8198	0.7554	0.8585	0.7932	0.9098	0.8521
	Coarse Dropout	0.8624	0.8071	0.8882	0.8260	0.8266	0.7631	0.8585	0.7930	0.9097	0.8517
	Elastic Transform	0.8722	0.8168	0.8913	0.8294	0.8281	0.7634	0.8553	0.7887	0.9113	0.8536
	Emboss	0.8659	0.8103	0.8876	0.8250	0.8237	0.7607	0.8552	0.7889	0.9113	0.8536
	Flip	0.8659	0.8103	0.8911	0.8290	0.8265	0.7628	0.8610	0.7958	0.9111	0.8535
	Gaussian Blur	0.8641	0.8089	0.8868	0.8242	0.8189	0.7555	0.8579	0.7925	0.9106	0.8527
	Grid Distortion	0.8687	0.8133	0.8922	0.8303	0.8272	0.7624	0.8605	0.7951	0.9115	0.8539
	Grid Dropout	0.8631	0.8069	0.8867	0.8243	0.8218	0.7576	0.8578	0.7917	0.9094	0.8511
	Image Compression	0.8620	0.8060	0.8870	0.8242	0.8203	0.7564	0.8589	0.7932	0.9101	0.8519
	Median Blur	0.8608	0.8055	0.8884	0.8259	0.8227	0.7584	0.8592	0.7933	0.9103	0.8526
	Optical Distortion	0.8685	0.8126	0.8898	0.8279	0.8236	0.7609	0.8608	0.7958	0.9110	0.8535
	Piecewise Affine	0.8708	0.8153	0.8915	0.8296	0.8318	0.7682	0.8709	0.8073	0.9119	0.8544
	Posterize	0.8643	0.8092	0.8897	0.8274	0.8261	0.7627	0.8641	0.7996	0.9110	0.8535
	RBC	0.8614	0.8054	0.8856	0.8226	0.8207	0.7569	0.8565	0.7905	0.9103	0.8522
	Random Crop	0.8607	0.8056	0.8870	0.8239	0.8195	0.7554	0.8524	0.7856	0.9088	0.8505
	Random Gamma	0.8675	0.8105	0.8854	0.8225	0.8192	0.7545	0.8526	0.7860	0.9087	0.8502
	Random Snow	0.8647	0.8096	0.8873	0.8244	0.8213	0.7580	0.8586	0.7931	0.9103	0.8527
	Rotate	0.8695	0.8144	0.8902	0.8279	0.8277	0.7635	0.8609	0.7956	0.9118	0.8543
	Sharpen	0.8661	0.8098	0.8901	0.8277	0.8167	0.7532	0.8545	0.7884	0.9108	0.8526
	Shift Scale Rotate	0.8680	0.8129	0.8910	0.8293	0.8289	0.7657	0.8647	0.8001	0.9115	0.8541
0.1	CLAHE	0.8561	0.8004	0.8866	0.8237	0.8172	0.7548	0.8606	0.7953	0.9084	0.8502
	Coarse Dropout	0.8609	0.8059	0.8883	0.8264	0.8239	0.7612	0.8648	0.8004	0.9103	0.8528
	Elastic Transform	0.8720	0.8176	0.8927	0.8313	0.8315	0.7676	0.8567	0.7904	0.9122	0.8552
	Emboss	0.8633	0.8083	0.8885	0.8263	0.8189	0.7564	0.8609	0.7958	0.9114	0.8539
	Flip	0.8691	0.8139	0.8907	0.8289	0.8267	0.7625	0.8542	0.7880	0.9104	0.8527
	Gaussian Blur	0.8657	0.8108	0.8880	0.8259	0.8134	0.7508	0.8576	0.7916	0.9098	0.8516
	Grid Distortion	0.8713	0.8173	0.8935	0.8317	0.8319	0.7685	0.8536	0.7870	0.9119	0.8545
	Grid Dropout	0.8610	0.8048	0.8873	0.8254	0.8297	0.7670	0.8668	0.8023	0.9108	0.8530
	Image Compression	0.8634	0.8081	0.8875	0.8250	0.8205	0.7570	0.8605	0.7950	0.9091	0.8511
	Median Blur	0.8627	0.8076	0.8890	0.8261	0.8155	0.7527	0.8561	0.7899	0.9100	0.8523
	Optical Distortion	0.8629	0.8085	0.8911	0.8291	0.8255	0.7615	0.8581	0.7925	0.9115	0.8539
	Piecewise Affine	0.8729	0.8180	0.8937	0.8320	0.8306	0.7666	0.8572	0.7910	0.9121	0.8547
	Posterize	0.8621	0.8074	0.8890	0.8265	0.8212	0.7576	0.8610	0.7958	0.9106	0.8526
	RBC	0.8617	0.8049	0.8885	0.8254	0.8241	0.7603	0.8608	0.7958	0.9098	0.8519
	Random Crop	0.8606	0.8050	0.8870	0.8239	0.8166	0.7536	0.8559	0.7899	0.9095	0.8513
	Random Gamma	0.8645	0.8091	0.8882	0.8253	0.8197	0.7550	0.8566	0.7906	0.9091	0.8510
	Random Snow	0.8616	0.8065	0.8896	0.8270	0.8227	0.7595	0.8576	0.7921	0.9106	0.8530
	Rotate	0.8724	0.8172	0.8924	0.8312	0.8353	0.7710	0.8568	0.7908	0.9117	0.8541
	Sharpen	0.8633	0.8080	0.8899	0.8278	0.8236	0.7603	0.8633	0.7985	0.9116	0.8545
	Shift Scale Rotate	0.8743	0.8204	0.8926	0.8317	0.8297	0.7653	0.8553	0.7890	0.9118	0.8544

Finally, for a probability of 0.3, the unified training set achieved a better F-score value in 9 data augmentation techniques in the MedSeg dataset, 16 in MosMed, 13 in Zenodo. In both probabilities of 0.25 and 0.3, all data augmentation techniques achieved a better F-score value in the CC-CCII dataset through the unified training strategy. The Ricord1a was the only dataset which did not achieved improvements with the unified training set. This happened due to the balancing approach that prioritized the small datasets that achieved poor results in the traditional training strategy when compared with the Ricord1a.

Moreover, data augmentation techniques were consistently more effective with this training strategy in all six probabilities evaluated. For a probability of 0.05, a higher F-score was achieved with Elastic Transform and Piecewise Affine on the CC-CCII dataset. In the MedSeg dataset, 7 data augmentation techniques achieved a higher F-score. In the MosMed, 12 data augmentation techniques achieved a higher F-score, with Piecewise Affine increasing the F-score by 2%. As presented in Section 3.2, applying data augmentation in the Zenodo dataset using the standard training sets did not

Table 7. Results of the data augmentation evaluation when unifying the training sets (Continuation of Table 6). p stands for probability, the blue-colored values indicate the best F-score values, and the red-colored values indicate the best IoU values. The values highlighted in green show the data augmentation techniques in which the P-value achieved values lower than 0.05, and thus the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation). The underscored values show the techniques where training with the combined training sets achieved a P-value lower than 0.05 when compared with training with a single training set, and the null hypothesis was rejected.

p	Augmentation	CC-CCII		MedSeg		MosMed		Ricord1a		Zenodo	
		F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU
	No Augmentation	0.8636	0.8087	0.8881	0.8253	0.8185	0.7547	0.8599	0.7947	0.9096	0.8514
0.15	CLAHE	0.8548	0.7992	0.8883	0.8252	0.8179	0.7552	0.8602	0.7947	0.9088	0.8506
	Coarse Dropout	0.8629	0.8071	0.8895	0.8276	0.8283	0.7642	0.8610	0.7959	0.9100	0.8522
	Elastic Transform	0.8734	0.8203	0.8920	0.8310	0.8345	0.7704	0.8594	0.7938	0.9111	0.8537
	Emboss	0.8643	0.8097	0.8908	0.8291	0.8229	0.7591	0.8577	0.7918	0.9115	0.8541
	Flip	0.8704	0.8164	0.8922	0.8308	0.8358	0.7717	0.8587	0.7933	0.9123	0.8547
	Gaussian Blur	0.8641	0.8086	0.8912	0.8295	0.8207	0.7575	0.8729	0.8100	0.9127	0.8557
	Grid Distortion	0.8736	0.8193	0.8939	0.8326	0.8353	0.7720	0.8612	0.7961	0.9136	0.8568
	Grid Dropout	0.8633	0.8080	0.8873	0.8250	0.8246	0.7610	0.8635	0.7987	0.9102	0.8521
	Image Compression	0.8654	0.8103	0.8863	0.8235	0.8199	0.7561	0.8574	0.7915	0.9095	0.8511
	Median Blur	0.8649	0.8093	0.8901	0.8286	0.8258	0.7615	0.8677	0.8036	0.9116	0.8545
	Optical Distortion	0.8676	0.8130	0.8914	0.8297	0.8245	0.7610	0.8538	0.7874	0.9114	0.8541
	Piecewise Affine	0.8758	0.8225	0.8944	0.8335	0.8353	0.7713	0.8610	0.7955	0.9135	0.8565
	Posterize	0.8623	0.8070	0.8905	0.8280	0.8242	0.7604	0.8617	0.7967	0.9097	0.8520
	RBC	0.8581	0.8019	0.8884	0.8255	0.8222	0.7577	0.8570	0.7910	0.9086	0.8505
	Random Crop	0.8648	0.8095	0.8890	0.8267	0.8238	0.7602	0.8668	0.8026	0.9109	0.8533
	Random Gamma	0.8609	0.8045	0.8887	0.8258	0.8176	0.7536	0.8596	0.7938	0.9090	0.8508
	Random Snow	0.8609	0.8063	0.8907	0.8285	0.8281	0.7651	0.8643	0.7997	0.9104	0.8531
	Rotate	0.8762	0.8219	0.8944	0.8333	0.8350	0.7696	0.8517	0.7847	0.9115	0.8539
	Sharpen	0.8641	0.8087	0.8893	0.8273	0.8220	0.7580	0.8663	0.8022	0.9128	0.8558
	Shift Scale Rotate	0.8766	0.8234	0.8941	0.8336	0.8374	0.7744	0.8652	0.8007	0.9144	0.8576
0.2	CLAHE	0.8569	0.8020	0.8872	0.8240	0.8183	0.7546	0.8602	0.7946	0.9093	0.8509
	Coarse Dropout	0.8597	0.8052	0.8873	0.8255	0.8244	0.7605	0.8575	0.7915	0.9106	0.8528
	Elastic Transform	0.8746	0.8215	0.8959	0.8352	0.8356	0.7728	0.8572	0.7912	0.9133	0.8564
	Emboss	0.8632	0.8088	0.8907	0.8288	0.8285	0.7641	0.8686	0.8046	0.9131	0.8557
	Flip	0.8722	0.8177	0.8926	0.8310	0.8316	0.7672	0.8561	0.7900	0.9119	0.8542
	Gaussian Blur	0.8615	0.8067	0.8888	0.8264	0.8170	0.7527	0.8616	0.7965	0.9116	0.8541
	Grid Distortion	0.8750	0.8219	0.8961	0.8356	0.8391	0.7761	0.8597	0.7941	0.9133	0.8561
	Grid Dropout	0.8613	0.8058	0.8879	0.8259	0.8240	0.7611	0.8591	0.7934	0.9090	0.8506
	Image Compression	0.8625	0.8068	0.8876	0.8247	0.8166	0.7537	0.8569	0.7910	0.9091	0.8510
	Median Blur	0.8656	0.8104	0.8901	0.8282	0.8173	0.7545	0.8630	0.7979	0.9108	0.8531
	Optical Distortion	0.8678	0.8133	0.8921	0.8301	0.8264	0.7633	0.8632	0.7984	0.9121	0.8550
	Piecewise Affine	0.8761	0.8231	0.8957	0.8351	0.8365	0.7732	0.8603	0.7948	0.9140	0.8569
	Posterize	0.8609	0.8058	0.8889	0.8264	0.8201	0.7559	0.8530	0.7860	0.9104	0.8523
	RBC	0.8599	0.8045	0.8873	0.8240	0.8209	0.7565	0.8486	0.7812	0.9086	0.8502
	Random Crop	0.8632	0.8070	0.8871	0.8243	0.8209	0.7576	0.8632	0.7986	0.9103	0.8523
	Random Gamma	0.8631	0.8074	0.8860	0.8233	0.8206	0.7572	0.8587	0.7929	0.9096	0.8516
	Random Snow	0.8632	0.8089	0.8909	0.8286	0.8278	0.7650	0.8622	0.7971	0.9116	0.8540
	Rotate	0.8753	0.8216	0.8949	0.8343	0.8391	0.7741	0.8542	0.7878	0.9121	0.8544
	Sharpen	0.8631	0.8077	0.8895	0.8277	0.8202	0.7566	0.8633	0.7985	0.9115	0.8541
	Shift Scale Rotate	0.8761	0.8230	0.8992	0.8392	0.8431	0.7808	0.8666	0.8021	0.9135	0.8568

improve the results; only one data augmentation improved the F-score. However, with this training strategy, 12 data augmentation techniques achieved a higher F-score on the Zenodo dataset. In the Ricord1a, four data augmentation techniques achieved a higher F-score, with Piecewise Affine also increasing the F-score by 2%.

For a probability of 0.1, 5 data augmentation techniques improved the F-score on the CC-CCII, 7 on the MedSeg, 13 on the MosMed, 4 on the Ricord1a, and 12 on the Zenodo. Also, in MosMed, the Elastic Transform, Grid Distortion, Piecewise Affine, and Rotate increased the F-score by 2%. For a probability of 0.15, 6 data augmentation methods improved the F-score on the CC-CCII, 12 on the MedSeg, 16 on the MosMed, 9 on the Ricord1a, and 13 on the Zenodo. In MosMed, six data augmentation techniques increased the F-score by 2%.

For a probability of 0.2, 5 data augmentation techniques improved the F-score on CC-CCII, 10 on MedSeg, 12 on the MosMed, 8 on the Ricord1a, and 13 on Zenodo. In MosMed, six data augmentation techniques increased the F-score by 2%, and the Shift Scale Rotate technique increased the F-score by 3%. For a probability of 0.25, 6

Table 8. Results of the data augmentation evaluation when unifying the training sets (Continuation of Table 6 and Table 7). p stands for probability, the blue-colored values indicate the best F-score values, and the red-colored values indicate the best IoU values. The values highlighted in green show the data augmentation techniques in which the P-value achieved values lower than 0.05, and thus the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation). The underscored values show the techniques in which training with the combined training sets achieved a P-value lower than 0.05 when compared with training with a single training set, and the null hypothesis was rejected.

p	Augmentation	CC-CCII		MedSeg		MosMed		Ricord1a		Zenodo	
		F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU
	No Augmentation	0.8636	0.8087	0.8881	0.8253	0.8185	0.7547	0.8599	0.7947	0.9096	0.8514
0.25	CLAHE	0.8565	0.8006	0.8856	0.8227	0.8212	0.7579	0.8654	0.8010	0.9098	0.8516
	Coarse Dropout	0.8635	0.8084	0.8899	0.8279	0.8292	0.7653	0.8637	0.7988	0.9113	0.8538
	Elastic Transform	0.8758	0.8235	0.8981	0.8376	0.8378	0.7739	0.8594	0.7937	0.9135	0.8568
	Emboss	0.8641	0.8105	0.8925	0.8306	0.8233	0.7600	0.8707	0.8074	0.9124	0.8555
	Flip	0.8717	0.8178	0.8926	0.8311	0.8338	0.7697	0.8546	0.7884	0.9113	0.8532
	Gaussian Blur	0.8604	0.8060	0.8898	0.8279	0.8218	0.7582	0.8682	0.8040	0.9115	0.8542
	Grid Distortion	0.8761	0.8236	0.8960	0.8349	0.8398	0.7759	0.8547	0.7882	0.9137	0.8568
	Grid Dropout	0.8628	0.8072	0.8890	0.8270	0.8270	0.7634	0.8607	0.7952	0.9079	0.8499
	Image Compression	0.8629	0.8078	0.8892	0.8269	0.8195	0.7569	0.8669	0.8029	0.9112	0.8535
	Median Blur	0.8622	0.8081	0.8881	0.8261	0.8219	0.7585	0.8632	0.7981	0.9123	0.8550
	Optical Distortion	0.8699	0.8161	0.8927	0.8317	0.8307	0.7681	0.8636	0.7987	0.9126	0.8560
	Piecewise Affine	0.8778	0.8250	0.8959	0.8357	0.8425	0.7791	0.8633	0.7982	0.9144	0.8577
	Posterize	0.8627	0.8068	0.8882	0.8259	0.8160	0.7521	0.8522	0.7857	0.9089	0.8506
	RBC	0.8602	0.8035	0.8882	0.8255	0.8228	0.7586	0.8602	0.7948	0.9098	0.8517
	Random Crop	0.8632	0.8072	0.8883	0.8257	0.8220	0.7587	0.8634	0.7990	0.9101	0.8526
	Random Gamma	0.8640	0.8087	0.8893	0.8272	0.8229	0.7600	0.8611	0.7957	0.9096	0.8518
	Random Snow	0.8634	0.8093	0.8913	0.8291	0.8288	0.7665	0.8665	0.8021	0.9130	0.8556
	Rotate	0.8762	0.8221	0.8952	0.8343	0.8409	0.7758	0.8514	0.7848	0.9118	0.8539
	Sharpen	0.8625	0.8074	0.8906	0.8282	0.8183	0.7549	0.8576	0.7919	0.9115	0.8540
	Shift Scale Rotate	0.8774	0.8244	0.8973	0.8372	0.8432	0.7799	0.8602	0.7945	0.9133	0.8560
0.3	CLAHE	0.8564	0.8001	0.8854	0.8229	0.8202	0.7569	0.8648	0.8000	0.9096	0.8516
	Coarse Dropout	0.8632	0.8084	0.8910	0.8290	0.8284	0.7645	0.8598	0.7945	0.9094	0.8517
	Elastic Transform	0.8742	0.8224	0.8977	0.8379	0.8422	0.7790	0.8613	0.7960	0.9146	0.8581
	Emboss	0.8657	0.8110	0.8901	0.8280	0.8219	0.7593	0.8682	0.8039	0.9127	0.8557
	Flip	0.8732	0.8193	0.8933	0.8323	0.8297	0.7658	0.8501	0.7831	0.9108	0.8523
	Gaussian Blur	0.8644	0.8085	0.8899	0.8281	0.8154	0.7530	0.8718	0.8085	0.9121	0.8548
	Grid Distortion	0.8750	0.8225	0.8994	0.8394	0.8448	0.7814	0.8625	0.7975	0.9141	0.8575
	Grid Dropout	0.8563	0.8019	0.8881	0.8262	0.8277	0.7646	0.8627	0.7973	0.9094	0.8511
	Image Compression	0.8629	0.8078	0.8882	0.8259	0.8243	0.7607	0.8639	0.7993	0.9115	0.8538
	Median Blur	0.8647	0.8093	0.8900	0.8274	0.8190	0.7550	0.8608	0.7954	0.9124	0.8552
	Optical Distortion	0.8700	0.8157	0.8928	0.8316	0.8238	0.7613	0.8637	0.7989	0.9141	0.8575
	Piecewise Affine	0.8764	0.8229	0.8952	0.8349	0.8384	0.7750	0.8598	0.7942	0.9137	0.8567
	Posterize	0.8625	0.8067	0.8902	0.8281	0.8243	0.7597	0.8593	0.7940	0.9111	0.8536
	RBC	0.8591	0.8028	0.8904	0.8281	0.8299	0.7653	0.8632	0.7982	0.9094	0.8514
	Random Crop	0.8631	0.8073	0.8859	0.8228	0.8193	0.7558	0.8659	0.8017	0.9107	0.8527
	Random Gamma	0.8635	0.8077	0.8883	0.8254	0.8199	0.7562	0.8574	0.7912	0.9092	0.8511
	Random Snow	0.8659	0.8112	0.8901	0.8284	0.8333	0.7703	0.8671	0.8029	0.9125	0.8557
	Rotate	0.8774	0.8237	0.8964	0.8356	0.8401	0.7757	0.8494	0.7823	0.9112	0.8533
	Sharpen	0.8635	0.8085	0.8890	0.8270	0.8144	0.7512	0.8629	0.7980	0.9113	0.8539
	Shift Scale Rotate	0.8767	0.8239	0.8984	0.8382	0.8417	0.7786	0.8583	0.7924	0.9126	0.8554

data augmentation techniques improved the F-score on CC-CCII, 10 on the MedSeg, 16 on MosMed, 10 on Ricord1a, and 15 on Zenodo. In the MosMed dataset, three data augmentation techniques increased the F-score by 3%.

Finally, for a probability of 0.3, 7 data augmentation techniques improved the F-score on CC-CCII, 12 on MedSeg, 14 on MosMed, 11 on Ricord1a, and 13 on Zenodo. In the MosMed dataset, four data augmentation techniques increased the F-score by 3%. In general, the probabilities of 0.25 and 0.3 achieved the highest number of data augmentations with improvements in the F-score values.

The best result per dataset was: CC-CCII with Piecewise Affine applied with a probability of 0.25, reaching an F-score of 0.8778; MedSeg with Grid Distortion applied with a probability of 0.3, achieving an F-score of 0.8994; MosMed with Grid Distortion applied with a probability of 0.3, attaining an F-score of 0.8448; Ricord1a with Gaussian Blur applied with a probability of 0.15, achieving an F-score of 0.8729; Zenodo with Elastic Transform applied with a probability of 0.3, reaching an F-score of 0.9146. Thus, in the CC-CCII, MedSeg, MosMed and Zenodo datasets the highest

F-score value was attained through a spatial transformation, while in Ricord1a it was achieved with a color transformation.

With the unified training set, the spatial operations also provided the best results, confirming that these operations are the best choices for the approached problem because they encourage shape variation of the lesion regions. The Ricord1a dataset is an exception, with the Gaussian Blur achieving the highest F-score value, suggesting that this dataset is more sensitive to color operations.

3.4. Experiment II: GAN-Based Data Augmentation

The datasets available for COVID-19 segmentation problems are limited in the number of images and have a critical imbalance problem; similar issues were identified in other medical images datasets (Banik and Bhattacharjee 2021; Sanagavarapu et al. 2021). This led to the next step of this work, which aims to develop and evaluate a domain-specific data augmentation technique based on GANs.

This evaluation is divided into three steps, as shown in Figure 2. First, we use and evaluate two GANs to generate healthy CT images. Second, we use an encoder-decoder segmentation network to generate the lung region segmentation mask of the healthy CT scans generated by the GAN models. Lastly, new lesions are added to the segmented lung regions, generating a new CT image with COVID-19 lesions.

3.4.1. Healthy CT Scans Generation

The first step of the proposed data augmentation strategy is generating synthetic healthy CT scan samples. To this end, two well-known GANs were evaluated: StarGAN v2 (Choi et al. 2020) and StyleGAN2-ADA (Karras et al. 2020). Regarding the StarGAN v2 model, it was trained for 1M iterations, with each iteration being the processing of a batch of size 4 through the network. Since StarGAN v2 is a GAN designed to perform image-to-image translation, two domains are required to train the network. In this case, the two domains are healthy lung images and COVID-19 infected lung images. To train the StarGAN v2 model, the images with COVID-19 infections were taken from the Ricord1a dataset, while the healthy ones were taken from Ricord1b.
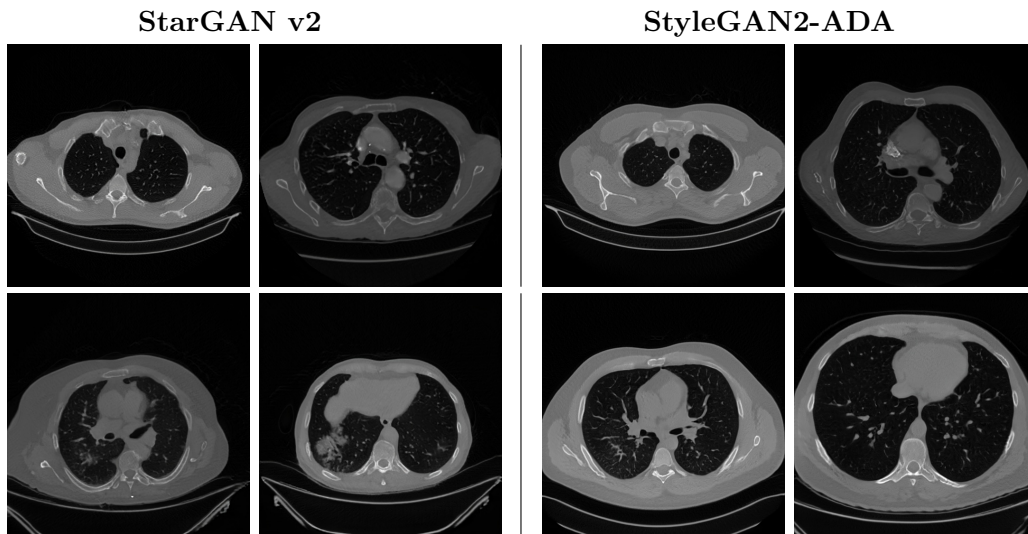

Figure 4. Examples of healthy lung images generated by StarGAN v2 and StyleGAN2-ADA.

16

Regarding the StyleGAN2-ADA model, it was trained 25 thousand *kimg*, a metric that counts the number of thousand images shown to the network (Karras et al. 2020). Unlike the StarGAN v2 model, StyleGAN2-ADA needed only one dataset to train the network. The dataset used in this step was the Ricord1b which provided healthy lung images. Figure 4 presents healthy images generated with the StarGAN v2 (on the left) and the StyleGAN2-ADA (on the right). The images generated with the StyleGAN2-ADA are also promising to be used in the next steps of the data augmentation framework. As presented above, both GANs achieved impressive results and generated fake healthy lung images close to the real healthy lung images. However, the StyleGAN2-ADA got a slightly better result, achieving an FID of 13.96, while the StarGAN v2 achieved an FID of 19.82. In general, the GANs achieved promising results and generated CT images close to the real examples from the datasets. The next step of the proposed data augmentation will use a segmentation network to generate a segmentation mask of the lung regions of the images generated by both GANs.

3.4.2. Lung Segmentation

In the second step of the proposed data augmentation technique, a segmentation model receives the images from the GAN models and generates the lung segmentation masks of these new images. The lung segmentation masks are explored to ensure that the new lesions are placed only within the lung region of the image. Based on the results reported in (Krinski et al. 2022), the encoder-decoder chosen in this step was the RegNetx-002 as the encoder and U-Net++ as the decoder. This combination presented impressive results in the experiments. Also, the RegNetx-002 is a relatively small network, making it faster for training and evaluation.

In order to train the network in this step, the CC-CCII and Zenodo datasets were combined. These datasets were the only two datasets with labels for the lung regions. This experiment also was validated through a 5-fold cross-validation strategy, and no data augmentation was applied.
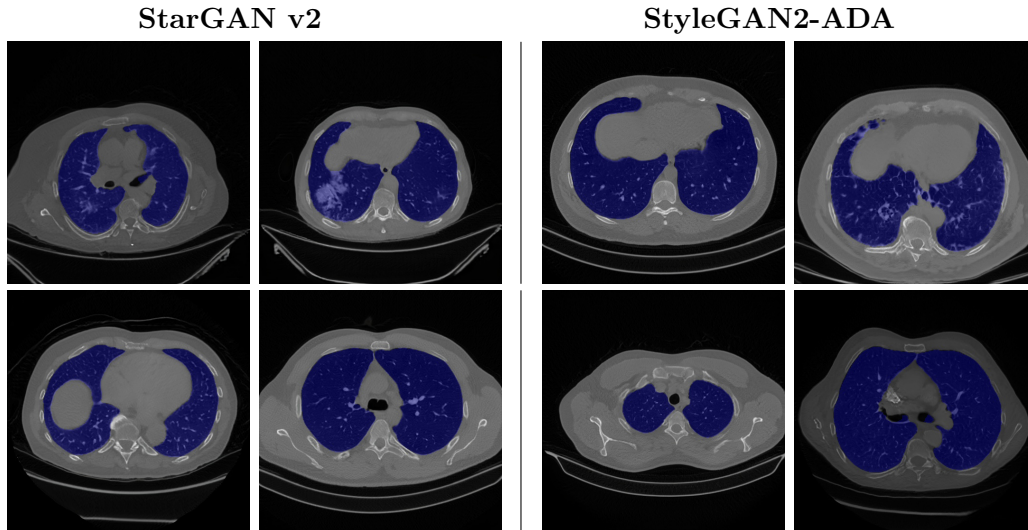

Figure 5. Examples of lung segmentation in the images generated by StarGAN v2 and StyleGAN2-ADA.

The images generated by the StarGAN v2 and StyleGAN2-ADA models were used as the test sets. That is, the segmentation network did not include images generated by the GANs in the training and validation steps. Figure 5 presents examples of lung seg-

mentation in the images generated by StarGAN v2 (left) and StyleGAN2-ADA (right). The blue-colored regions are the lung regions segmented by the network. The test sets have no ground-truth labels for quantitative analysis. However, the segmentation network achieved promising qualitative results. The generated segmentation masks are close to the expected if made through a manual labeling process, suggesting that the generated samples are visually close enough to the original images.

3.4.3. Adding New Lesions

The next step is the addition of COVID-19 lesions within the lung regions of the CT scans (see Section 3.4.1) generated by the GANs (see Section 3.4.2). For each random healthy image generated by a GAN model (StarGAN v2 or StyleGAN2-ADA), a random image with lesions is chosen from the dataset Ricord1a. The first problem in this step is that the size of the lungs inside the CT scans varies greatly. Therefore, a matching step is performed to check if the size of the lung region of the image from the dataset is close to the size of the lung region of the healthy image generated by the GAN model. A threshold of 10% was used, i.e., the size of the lungs of the lesion image can be at a maximum of 10% greater or 10% smaller than the GAN image.
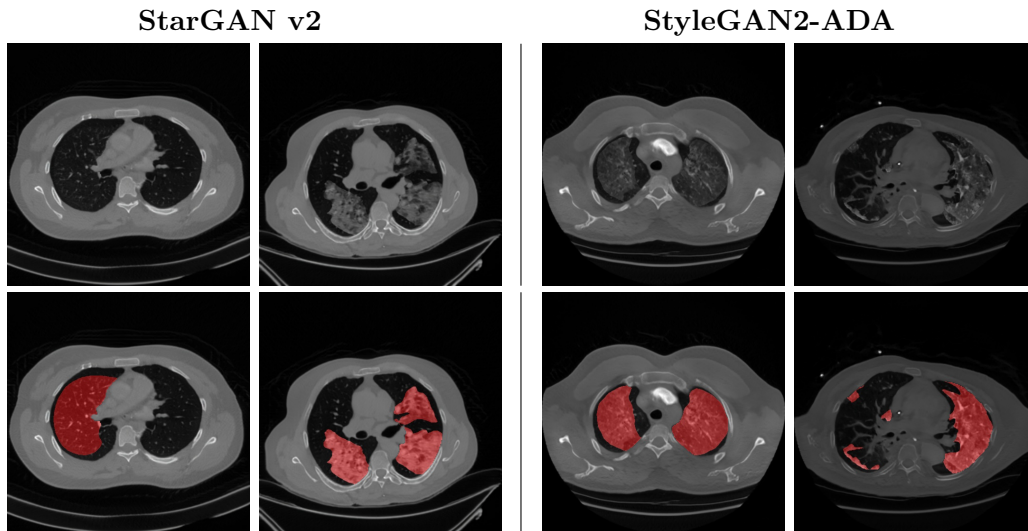
StarGAN v2   **StyleGAN2-ADA**

Figure 6. Examples of images generated by StarGAN v2 and StyleGAN2-ADA after adding COVID-19 lesions and the corresponding generated segmentation masks.

After matching the lesion image from the dataset with the image generated by the GAN model, the lungs of the dataset image are cropped from the original image and positioned on top of the GAN image. The center of the bounding box of the lungs is calculated and used to align the lungs of the dataset image with the lungs of the GAN image. Then, the functions AddWeighted and GaussianBlur from the OpenCV Library (Bradski 2000) are applied to mix the images and smooth out the edges, respectively. Figure 6 illustrates the resulting images and the corresponding segmentation masks with lesions added to images generated by StarGAN v2 (left) and StyleGAN2-ADA (right). As expected, due to the promising lung segmentation step presented in Section 3.4.2, the COVID-19 lesions were correctly placed within the lung regions.

3.4.4. Proposed GAN-Based Data Augmentation Evaluation

Table 9 (a continuation of Tables 6, 7 and 8) presents the evaluation results for the proposed data augmentation technique in the unified training set. It was evaluated with two variations (with and without flip). First, the lesions from the dataset were added to the healthy lung images generated by the GANs without altering the lesion images. Then, in the second variation, the lesion images were horizontally flipped before being added to the healthy image generated by the GANs.

Table 9. Results of the data augmentation evaluation when unifying the training sets (Continuation of Tables 6, 7 and 8. p stands for probability, the blue-colored values indicate the best F-score values, and the red-colored values indicate the best IoU values. The values highlighted in green show the data augmentation techniques in which the P-value achieved values lower than 0.05, and thus the null hypothesis was rejected (i.e., there is a statistical difference and the results achieved are better than without data augmentation). Two variations of the proposed algorithm were evaluated. The + F indicates the variation in which the lesion images were horizontally flipped before being added to the healthy image generated by the GANs.

p	Augmentation	CC-CCII		MedSeg		MosMed		Ricord1a		Zenodo	
		F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU	F-score	IoU
	No Augmentation	0.8636	0.8087	0.8881	0.8253	0.8185	0.7547	0.8599	0.7947	0.9096	0.8514
0.05	StarGAN v2	0.8615	0.8060	0.8893	0.8270	0.8200	0.7565	0.8653	0.8009	0.9107	0.8533
	StarGAN v2 + F	0.8628	0.8071	0.8873	0.8247	0.8193	0.7566	0.8620	0.7974	0.9106	0.8528
	StyleGAN2-ADA	0.8624	0.8071	0.8861	0.8235	0.8180	0.7546	0.8535	0.7872	0.9076	0.8491
	StyleGAN2-ADA + F	0.8644	0.8085	0.8873	0.8246	0.8163	0.7534	0.8592	0.7938	0.9098	0.8518
0.1	StarGAN v2	0.8632	0.8082	0.8853	0.8222	0.8133	0.7507	0.8560	0.7908	0.9082	0.8499
	StarGAN v2 + F	0.8623	0.8058	0.8871	0.8244	0.8176	0.7545	0.8627	0.7980	0.9091	0.8508
	StyleGAN2-ADA	0.8617	0.8061	0.8879	0.8251	0.8213	0.7585	0.8672	0.8034	0.9117	0.8540
	StyleGAN2-ADA + F	0.8616	0.8058	0.8891	0.8268	0.8227	0.7598	0.8711	0.8081	0.9102	0.8527
0.15	StarGAN v2	0.8637	0.8087	0.8870	0.8243	0.8214	0.7586	0.8689	0.8055	0.9099	0.8521
	StarGAN v2 + F	0.8654	0.8096	0.8871	0.8248	0.8239	0.7610	0.8665	0.8031	0.9107	0.8527
	StyleGAN2-ADA	0.8645	0.8086	0.8872	0.8239	0.8155	0.7528	0.8612	0.7965	0.9090	0.8509
	StyleGAN2-ADA + F	0.8622	0.8066	0.8885	0.8260	0.8215	0.7586	0.8696	0.8062	0.9115	0.8541
0.2	StarGAN v2	0.8644	0.8085	0.8860	0.8231	0.8201	0.7563	0.8555	0.7901	0.9084	0.8499
	StarGAN v2 + F	0.8655	0.8098	0.8867	0.8242	0.8192	0.7556	0.8626	0.7983	0.9097	0.8518
	StyleGAN2-ADA	0.8635	0.8093	0.8886	0.8256	0.8190	0.7558	0.8607	0.7959	0.9103	0.8522
	StyleGAN2-ADA + F	0.8614	0.8065	0.8895	0.8269	0.8201	0.7571	0.8729	0.8103	0.9104	0.8528
0.25	StarGAN v2	0.8660	0.8109	0.8891	0.8264	0.8189	0.7562	0.8664	0.8029	0.9097	0.8518
	StarGAN v2 + F	0.8643	0.8081	0.8893	0.8262	0.8192	0.7558	0.8693	0.8061	0.9090	0.8510
	StyleGAN2-ADA	0.8617	0.8060	0.8878	0.8251	0.8183	0.7548	0.8619	0.7976	0.9101	0.8520
	StyleGAN2-ADA + F	0.8668	0.8116	0.8875	0.8245	0.8222	0.7583	0.8656	0.8017	0.9099	0.8520
0.3	StarGAN v2	0.8635	0.8078	0.8851	0.8221	0.8204	0.7572	0.8617	0.7975	0.9088	0.8504
	StarGAN v2 + F	0.8645	0.8087	0.8884	0.8259	0.8214	0.7584	0.8737	0.8111	0.9105	0.8532
	StyleGAN2-ADA	0.8648	0.8100	0.8877	0.8247	0.8197	0.7562	0.8620	0.7975	0.9095	0.8512
	StyleGAN2-ADA + F	0.8675	0.8119	0.8876	0.8247	0.8195	0.7568	0.8721	0.8095	0.9105	0.8530

Also, unlike the previously evaluated data augmentation technique, the proposed one is not applicable in an online manner due to the several steps needed to generate the new images. Thus, the proposed data augmentation method was evaluated through a technique called offline augmentation. In this technique, the images are generated before the network training, increasing the number of training images. The number of images was increased by 0.05, 0.1, 0.15, 0.2, 0.25, and 0.30 to preserve the same probabilities evaluated previously. The proposed data augmentation method achieved better F-scores in three datasets: MosMed, Ricord1a, and Zenodo. Also, in Ricord1a, our method achieved the highest F-score of all evaluated techniques in the unified training strategy applied with probabilities 0.1, 0.2, and 0.3. The proposed data augmentation method generally achieved close results with the images generated with the StarGAN v2 and StyleGAN2-ADA. Additionally, the version where the healthy lung images were flipped before receiving a lesion achieved close results to the version without flipping.

It is important to highlight that our proposed approach produced the highest result on the dataset Ricord1a with the unified training strategy, which we previously indi-
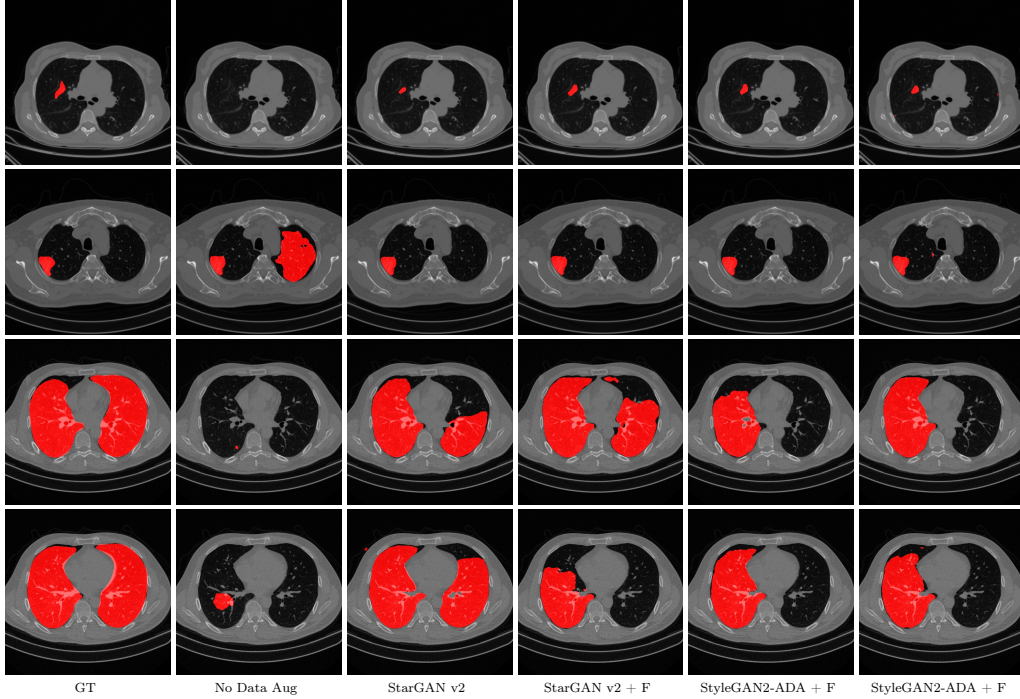
Figure 7. Qualitative comparison between the proposed data augmentation and the baseline without data augmentation. With the proposed data augmentation, the segmentation network found the lesion region closer to the expected in the ground truth.

cated as more sensitive to color changes. The proposed approach provides a variety of both cases: color and shape specifically adapted for this problem, although there is a greater focus on color. Our hypothesis is that inserting real lesions (a proportionally small quantity of pixels) on completely artificial backgrounds (the majority of the pixels) encourages a higher variation of both color and shape instead of just one of them, which proved more beneficial for this particular dataset than the traditional approaches that focus on only one aspect.

Although it did not achieve the highest F-score value, our approach improved the F-score on the MosMed and Zenodo datasets and achieved competitive results to the traditional approaches. In CC-CCII and MedSeg, our approach did not outperform the baseline without data augmentation. This occurred because the GANs were trained with the Ricord1b dataset, and the background regions are different from those datasets. Including background regions close to those datasets was left for future work.

Moreover, in some cases, the new samples drastically improved the mask quality, as seen in Figure 7. In the first row, without data augmentation, the segmentation network could not even find any lesion, a failure case with a severe biological implication. In the example presented in the second row, without data augmentation, the segmentation network presents a meager recall value, predicting a significant false positive area. In the third and fourth rows, there is a low precision, with many false negative pixels, thus a low F-score value, on an image where the lesions almost wholly cover the lungs.

In medical problems, false negative results are generally the worst scenario, highly increasing patient risks (Woloshin et al. 2020; Wikramaratna et al. 2020; Kanji et al. 2021). In such examples, the proposed data augmentation approach helps the network find a segmentation mask closer to the ground truth, avoiding a false positive result.

Furthermore, as presented in Figure 7, the proposed data augmentation method improved the segmentation task by finding a segmentation mask closer to the expected.

In Figure 7, note that the four variations analyzed of our proposed technique managed to produce much better results with the same network architecture than those without data augmentation. We believe this is due to the small number of images in the original datasets. Thus, the diversity introduced by the proposed technique improved the generalization in some cases.

4. Conclusion and Future Work

In this work, we evaluated 24 data augmentation techniques on COVID-19 CT scans on five datasets using an encoder-decoder network composed of RegNetx-002 (Xu et al. 2022b) and U-net++ (Zhou et al. 2018), comparing six different probabilities of applying the techniques 0.05, 0.1, 0.15, 0.2, 0.25, and 0.3. To the best of our knowledge, this is by far the most extensive evaluation done on this topic. Two limitations of our work are: (i) parameter optimization was left out; and (ii) although six discrete probabilities were evaluated, the results indicate that there may be an optimal probability outside the range $[0.05, 0.3]$. Hence, more experiments should be conducted in future works.

In addition to the traditional approach, where the training and test sets are disjoint subsets from the same dataset, we proposed a different methodology where the training subsets are combined into a single larger set. We showed that the data augmentation techniques were consistently more effective (i.e., they reached better F-score and IoU values) with this training strategy in four of the datasets we performed experiments. The Ricord1a was the only dataset which did not achieved improvements with the unified training set. This happened due to the balancing approach that prioritized the small datasets that achieved poor results in the traditional training strategy when compared with the Ricord1a. Also, although applying data augmentation techniques with a probability of 0.3 in the first evaluation did not show interesting results, the same probability produced the highest number of data augmentation techniques that improved the F-score when the datasets were combined into a single training set. Furthermore, the slight difference achieved by the data augmentation process in the individual training sets resulted in a small difference between the probabilities applied, making it unclear which is the optimal probability in this case. Meanwhile, applying data augmentation in the unified training set achieved overall higher results, clarifying that increasing the probability of the data augmentation techniques generates better results, opposing previously works from literature (Müller et al. 2020, 2021). Lastly, by using a higher image resolution than our previous work (Krinski et al. 2022) we obtained a significant gain in F-score in four of the datasets evaluated.

Additionally, our results show that the best operations in both training strategies are those that change shape instead of colors, such as Grid Distortion, Optical Distortion, Flip, Piecewise Affine, and Shift Scale Rotate. According to our results, data augmentation techniques like CLAHE, Coarse and Grid Dropout, Random Crop, Image Compression, Random Gamma, and Random Snow did not generate improved results and thus do not need to be applied to this problem.

Finally, we proposed a novel data augmentation technique that first employs a GAN model to produce new CT scans of healthy lungs and then combines existing labeled lesions with those new images to generate new samples and boost the segmentation performance. According to our results, this technique is promising, managing to improve the segmentation in certain cases. The StarGAN v2 + F with a probability 0.3

achieved the highest F-score on the Ricord1a dataset in the unified training strategy. However, there is still room for a more consistent improvement in future works. The generated images are similar to the real images with a low FID, and the trained segmentation models managed to generate a reasonable ground truth annotation for those new samples. This presents an opportunity for future works to improve our proposed data augmentation focusing on the COVID-19 CT segmentation problem but also opens the possibility of applying our pipeline to other medical problems with lesions inside the lung regions. The code used to perform our experiments is publicly available at `https://github.com/VRI-UFPR/DACov2022`.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Notes on contributors

Bruno Krinski is a PhD student at the Federal University of Paraná (UFPR), where he also received his master's degree in Computer Science (2019). He received his bachelor's degree in Computer Science (2016) from the UFPR. His research interests include deep learning, image processing, and computer vision.

Daniel V. Ruiz is a Master of Science (M.S) in Informatics at the Federal University of Paraná (UFPR) (2021). B.S. Computer Science at UFPR (2017). Exchange student via the CAPES-UNIBRAL program, an exchange program associated with the program Science without Borders, at the Friedrich-Alexander-Universität Erlangen-Nürnberg (2015-2016). Working fields: robotics, computer vision, machine learning, computer graphics, digital image processing, and web development.

Rayson Laroca received the bachelor's degree in software engineering from the State University of Ponta Grossa (UEPG), Brazil, and the master's degree in computer science from the Federal University of Paraná (UFPR), Brazil. He is currently a PhD student with UFPR. His research interests include computer vision, machine learning, and pattern recognition.

Eduardo Todt is an Electrical Engineering (1985) and Master in Computer Science (1990) from the Federal University of Rio Grande do Sul (UFRGS) and PhD in Advanced Automation and Robotics - Polytechnic University of Catalonia (IRI-UPC, 2005), in partnership with FZI Institute - Karlsruhe, obtaining the title of European Doctor, with Honor. Since 2008, professor at the Department of Informatics of the Federal University of Paraná (UFPR), coordinator of the Computer Science course from 2010 to 2014 and vice-coordinator of the Biomedical Informatics since january

2021. Coordinator of large scale projects related to Educational Linux, Educational Objects Platform, and dahsboard of the National Program of Books and Didatical Material (PNLD). Member of research groups Center of Scientific Computation and Free Software (C3SL), Vision, Robotics and Images (VRI), and Sustainable Traffic and Transport (TTS) at UFPR. Experience in Electrical Engineering and Computer Engineering, developing products and systems for industrial automation, as well as R&D manager. Main interests are on mobile robotics, computational vision and computer science in education. Member of the Steering Committee of the Special Commission on Robotics (CER) of the Brazilian Computer Society (SBC) since 2018.

References

Ai T, Yang Z, Hou H, Zhan C, Chen C, Lv W, Tao Q, Sun Z, Xia L. 2020. Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: A report of 1014 cases. Radiology. 296(2):E32–E40.

Anthimopoulos M, Christodoulidis S, Ebner L, Christe A, Mougiakakou S. 2016. Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. IEEE Transactions on Medical Imaging. 35(5):1207–1216.

Banik D, Bhattacharjee D. 2021. Mitigating data imbalance issues in medical image analysis. IGI Global. chap. 4; p. 66–89.

Bradski G. 2000. The OpenCV Library. Dr Dobb's Journal of Software Tools.

Buslaev A, Iglovikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA. 2020. Albumentations: Fast and flexible image augmentations. Information. 11(2).

Cao F, Bao Q. 2020. A survey on image semantic segmentation methods with convolutional neural network. In: International Conference on Communications, Information System and Computer Engineering (CISCE). p. 458–462.

Cao K, Bi L, Feng D, Kim J. 2020. Improving PET-CT image segmentation via deep multi-modality data augmentation. In: Machine learning for medical image reconstruction. Springer International Publishing; p. 145–152.

Chen C, Dou Q, Chen H, Qin J, Heng PA. 2019. Synergistic image and feature adaptation: Towards cross-modality domain adaptation for medical image segmentation. Proceedings of the AAAI Conference on Artificial Intelligence. 33(01):865–872.

Chen H, Jiang Y, Ko H, Loew M. 2023. A teacher–student framework with Fourier Transform augmentation for COVID-19 infection segmentation in CT images. Biomedical Signal Processing and Control. 79:104250.

Chen M, Tu C, Tan C, Zheng X, Wang X, Wu J, Huang Y, Wang Z, Yan Y, Li Z, et al. 2020a. Key to successful treatment of COVID-19: accurate identification of severe risks and early intervention of disease progression. medRxiv.

Chen P, Liu S, Zhao H, Jia J. 2020b. Gridmask data augmentation. arXiv preprint. 2001.04086.

Chen X, Yao L, Zhang Y. 2020c. Residual attention U-Net for automated multi-class segmentation of COVID-19 chest CT images. arXiv preprint. 2004.05645.

Choi Y, Uh Y, Yoo J, Ha JW. 2020. StarGAN v2: Diverse image synthesis for multiple domains. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). p. 8185–8194.

Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B. 2016. The Cityscapes dataset for semantic urban scene understanding. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). p. 3213–3223.

Demšar J. 2006. Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res. 7:1–30.

Diniz JOB, Quintanilha DBP, Neto ACS, da Silva GLF, Ferreira JL, Netto SMB, Araújo JDL, Cruz LBD, Silva TFB, da S Martins CM, et al. 2021. Segmentation and quantification of COVID-19 infections in CT using pulmonary vessels extraction and deep learning. Multi-

media Tools and Applications. 80(19):29367–29399.

Dmitriev K, Kaufman AE. 2019. Learning multi-class segmentations from single-class datasets. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); Jun. p. 9493–9503.

Dong N, Kampffmeyer M, Liang X, Wang Z, Dai W, Xing E. 2018. Unsupervised domain adaptation for automatic estimation of cardiothoracic ratio. In: International conference on medical image computing and computer-assisted intervention. Springer. p. 544–552.

Dunford R, Su Q, Tamang E. 2014. The pareto principle. The Plymouth Student Scientist. 7:140–148.

Enshaei N, Oikonomou A, Rafiee MJ, Afshar P, Heidarian S, Mohammadi A, Plataniotis KN, Naderkhani F. 2022. COVID-rate: an automated framework for segmentation of COVID-19 lesions from chest CT images. Scientific Reports. 12(1):3212.

Fung DLX, Liu Q, Zammit J, Leung CKS, Hu P. 2021. Self-supervised deep learning model for COVID-19 lung CT image segmentation highlighting putative causal relationship among age, underlying disease and COVID-19. Journal of Translational Medicine. 19(1).

Gite S, Mishra A, Kotecha K. 2022. Enhanced lung image segmentation using deep learning. Neural Computing and Applications.

Habili N, Kwan E, Li W, Webers C, Oorloff J, Armin MA, Petersson L. 2022. A hyperspectral and RGB dataset for building facade segmentation. arXiv preprint. 2212.02749.

Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S. 2017. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: International Conference on Neural Information Processing Systems (NeurIPS). p. 6629–6640.

Huang C, Wang Y, Li X, Ren L, Zhao J, Hu Y, Zhang L, Fan G, Xu J, Gu X, et al. 2020. Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. The Lancet. 395(10223):497–506.

Jaccard P. 1912. The distribution of the flora in the alpine zone. New Phytologist. 11(2):37–50.

Jaderberg M, Simonyan K, Zisserman A, Kavukcuoglu K. 2015. Spatial transformer networks. In: International Conference on Neural Information Processing Systems (NeurIPS). p. 2017–2025.

Jiang Y, Chen H, Loew M, Ko H. 2021. COVID-19 CT image synthesis with a conditional generative adversarial network. IEEE Journal of Biomedical and Health Informatics. 25(2):441–452.

Jiwani A, Ganguly S, Ding C, Zhou N, Chan DM. 2021. A semantic segmentation network for urban-scale building footprint extraction using RGB satellite imagery. arXiv preprint. 2104.01263.

Johns Hopkins University School of Medicine. 2022. Coronavirus resource center; [`https://coronavirus.jhu.edu/`]. Accessed: 2022-08-12.

Johnson JM, Khoshgoftaar TM. 2019. Survey on deep learning with class imbalance. Journal of Big Data. 6(1):27.

Jun M, Cheng G, Yixin W, Xingle A, Jiantao G, Ziqi Y, Minqing Z, Xin L, Xueyuan D, Shucheng C, et al. 2020. Covid-19 ct lung and infection segmentation dataset. Available from: `https://zenodo.org/record/3757475`.

Kaissis G, Ziller A, Passerat-Palmbach J, Ryffel T, Usynin D, Trask A, Lima I, Mancuso J, Jungmann F, Steinborn MM, et al. 2021. End-to-end privacy preserving deep learning on multi-institutional medical imaging. Nature Machine Intelligence. 3(6):473–484.

Kanji JN, Zelyas N, MacDonald C, Pabbaraju K, Khan MN, Prasad A, Hu J, Diggle M, Berenger BM, Tipples G. 2021. False negative rate of COVID-19 PCR testing: a discordant testing analysis. Virology Journal. 18(1):13.

Kapoor P, Bui TD. 2021. TinyStarGAN v2: Distilling StarGAN v2 for efficient diverse image synthesis for multiple domains. In: British Machine Vision Conference (BMVC).

Karras T, Aittala M, Hellsten J, Laine S, Lehtinen J, Aila T. 2020. Training generative adversarial networks with limited data. In: International Conference on Neural Information Processing Systems (NeurIPS). p. 1–11.

Krinski B, Ruiz D, Todt E. 2022. Light in the black: An evaluation of data augmentation tech-

niques for covid-19 ct's semantic segmentation. In: *XXII Simpósio Brasileiro de Computação Aplicada à Saúde*; Porto Alegre, RS, Brasil. p. 156–167.

Krinski BA, Ruiz DV, Todt E. 2021. Spark in the dark: Evaluating encoder-decoder pairs for covid-19 ct's semantic segmentation. In: Latin American Robotics Symposium (LARS). p. 198–203.

Kupyn O, Martyniuk T, Wu J, Wang Z. 2019. DeblurGAN-v2: Deblurring (orders-of-magnitude) faster and better. In: IEEE/CVF International Conference on Computer Vision (ICCV). p. 8877–8886.

Laroca R, Araujo AB, Zanlorensi LA, De Almeida EC, Menotti D. 2021. Towards image-based automatic meter reading in unconstrained scenarios: A robust and efficient approach. IEEE Access. 9:67569–67584.

Laroca R, Cardoso EV, Lucio DR, Estevam V, Menotti D. 2022a. On the cross-dataset generalization in license plate recognition. In: International Conference on Computer Vision Theory and Applications (VISAPP); Feb. p. 166–178.

Laroca R, Santos M, Estevam V, Luz E, Menotti D. 2022b. A first look at dataset bias in license plate recognition. In: Conference on Graphics, Patterns and Images; Oct. p. 234–239.

Li S, Cai H, Qi L, Yu Q, Shi Y, Gao Y. 2022. PLN: Parasitic-like network for barely supervised medical image segmentation. IEEE Transactions on Medical Imaging:1–1.

Liu Y, Shu Z, Li Y, Lin Z, Perazzi F, Kung SY. 2021. Content-aware gan compression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. p. 12156–12166.

Mahapatra D, Singh A. 2021. CT image synthesis using weakly supervised segmentation and geometric inter-label relations for covid image analysis. arXiv preprint. 2106.10230.

Mahmud T, Alam MJ, Chowdhury S, Ali SN, Rahman MM, Anowarul Fattah S, Saquib M. 2021. Covtanet: A hybrid tri-level attention-based network for lesion segmentation, diagnosis, and severity prediction of covid-19 chest ct scans. IEEE Transactions on Industrial Informatics. 17(9):6489–6498.

MedSeg. 2021. COVID-19 CT segmentation dataset; [http://medicalsegmentation.com/covid19/]. Accessed: 2021-05-03.

Minaee S, Boykov Y, Porikli F, Plaza A, Kehtarnavaz N, Terzopoulos D. 2022. Image segmentation using deep learning: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence. 44(7):3523–3542.

Mirza M, Osindero S. 2014. Conditional generative adversarial nets. arXiv preprint. arXiv:1411.1784:1–7.

Morozov S, Andreychenko A, Pavlov N, Vladzymyrskyy A, Ledikhova N, Gombolevskiy V, Blokhin I, Gelezhe P, Gonchar A, Chernina V. 2020. MosMedData: Chest CT scans with COVID-19 related findings dataset. medRxiv.

Müller D, Soto-Rey I, Kramer F. 2020. Automated chest CT image segmentation of COVID-19 lung infection based on 3D U-Net. arXiv preprint. arXiv: 2007.04774.

Müller D, Soto-Rey I, Kramer F. 2021. Robust chest CT image segmentation of COVID-19 lung infection based on limited data. Informatics in Medicine Unlocked. 25:100681.

Narin A, Kaya C, Pamuk Z. 2021. Automatic detection of coronavirus disease (COVID-19) using x-ray images and deep convolutional neural networks. Pattern Analysis and Applications. 24(3):1207–1220.

Pandey P, Tyagi AK, Ambekar S, Prathosh AP. 2020. Unsupervised domain adaptation for semantic segmentation of NIR images through generative latent search. In: European Conference on Computer Vision (ECCV). p. 413–429.

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. Red Hook, NY, USA: Curran Associates Inc.

Powers DMW. 2020. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. arXiv preprint. arXiv:2010.16061.

Qiblawey Y, Tahir A, Chowdhury MEH, Khandakar A, Kiranyaz S, Rahman T, Ibtehaz N, Mahmud S, Maadeed SA, Musharavati F, et al. 2021. Detection and severity classification

of covid-19 in ct images using deep learning. Diagnostics. 11(5).

Raj ANJ, Zhu H, Khan A, Zhuang Z, Yang Z, Mahesh VGV, Karthik G. 2021. ADID-UNET—a segmentation model for COVID-19 infection from lung CT scans. PeerJ Computer Science. 7:e349.

Ruiz DV, Krinski BA, Todt E. 2019. ANDA: A novel data augmentation technique applied to salient object detection. In: International Conference on Advanced Robotics. p. 487–492.

Ruiz DV, Krinski BA, Todt E. 2020. IDA: Improved data augmentation applied to salient object detection. In: Conference on Graphics, Patterns and Images. p. 210–217.

Saeedizadeh N, Minaee S, Kafieh R, Yazdani S, Sonka M. 2021. COVID TV-Unet: segmenting COVID-19 chest CT images using connectivity imposed Unet. Computer Methods and Programs in Biomedicine Update. 1:100007.

Salama WM, Aly MH. 2022. Framework for COVID-19 segmentation and classification based on deep learning of computed tomography lung images. Journal of Electronic Science and Technology. 20(3):100161. Available from: https://doi.org/10.1016/j.jnlest.2022.100161.

Sanagavarapu S, Sridhar S, Gopal T. 2021. COVID-19 identification in CLAHE enhanced CT scans with class imbalance using ensembled resnets. In: IEEE International IOT, Electronics and Mechatronics Conference. p. 1–7.

Saood A, Hatem I. 2021. COVID-19 lung CT image segmentation using deep learning methods: U-net versus SegNet. BMC Medical Imaging. 21(1).

Shi F, Wang J, Shi J, Wu Z, Wang Q, Tang Z, He K, Shi Y, Shen D. 2021. Review of artificial intelligence techniques in imaging data acquisition, segmentation, and diagnosis for COVID-19. IEEE Reviews in Biomedical Engineering. 14:4–15.

Sun W, Chen J, Yan L, Lin J, Pang Y, Zhang G. 2022. COVID-19 CT image segmentation method based on swin transformer. Frontiers in Physiology. 13.

Tsai E, Simpson S, Lungren MP, Hershman M, Roshkovan L, Colak E, Erickson BJ, Shih G, Stein A, Kalpathy-Cramer J, et al. 2020. Medical imaging data resource center - rsna international covid radiology database release 1a - chest ct covid+ (midrc-ricord-1a).

Urooj A, Borji A. 2018. Analysis of hand segmentation in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); June.

Wang C, Horby PW, Hayden FG, Gao GF. 2020. A novel coronavirus outbreak of global health concern. The Lancet. 395(10223):470–473.

Wikramaratna PS, Paton RS, Ghafari M, Lourenço J. 2020. Estimating the false-negative test probability of SARS-CoV-2 by RT-PCR. Eurosurveillance. 25(50).

Woloshin S, Patel N, Kesselheim AS. 2020. False negative tests for SARS-CoV-2 infection — challenges and implications. New England Journal of Medicine. 383(6):e38.

Xu G, Hou Y, Liu Z, Loy CC. 2022a. Mind the gap in distilling StyleGANs. In: European Conference on Computer Vision. Springer. p. 423–439.

Xu J, Pan Y, Pan X, Hoi S, Yi Z, Xu Z. 2022b. Regnet: Self-regulated network for image classification. IEEE Transactions on Neural Networks and Learning Systems:1–6.

Xu Z, Cao Y, Jin C, Shao G, Liu X, Zhou J, Shi H, Feng J. 2020. Gasnet: Weakly-supervised framework for covid-19 lesion segmentation. arXiv preprint. 2010.09456.

yan Yao H, gen Wan W, Li X. 2022. A deep adversarial model for segmentation-assisted COVID-19 diagnosis using CT images. EURASIP Journal on Advances in Signal Processing. 2022(1). Available from: https://doi.org/10.1186/s13634-022-00842-x.

Yazdekhasty P, Zindari A, Nabizadeh-ShahreBabak Z, Khadivi P, Karimi N, Samavi S. 2021. Segmentation of lungs covid infected regions by attention mechanism and synthetic data. arXiv preprint. 2108.08895.

Zhang J, Ding X, Hu D, Jiang Y. 2022. Semantic segmentation of COVID-19 lesions with a multiscale dilated convolutional network. Scientific Reports. 12(1).

Zhang K, Liu X, Shen J, Li Z, Sang Y, Wu X, Zha Y, Liang W, Wang C, Wang K, et al. 2020. Clinically applicable AI system for accurate diagnosis, quantitative measurements, and prognosis of COVID-19 pneumonia using computed tomography. Cell. 181(6):1423–1433.e11.

Zhao X, Zhang P, Song F, Fan G, Sun Y, Wang Y, Tian Z, Zhang L, Zhang G. 2021. D2A U-Net: Automatic segmentation of COVID-19 lesions from CT slices with dilated convolution and dual attention mechanism. arXiv preprint. 2102.05210.

Zhong Z, Zheng L, Kang G, Li S, Yang Y. 2020. Random erasing data augmentation. Proceedings of the AAAI Conference on Artificial Intelligence. 34(07):13001–13008.

Zhou Z, Siddiquee MMR, Tajbakhsh N, Liang J. 2018. UNet++: A nested U-Net architecture for medical image segmentation. In: International Workshop on Deep Learning in Medical Image Analysis. p. 3–11.