Application of Generative Adversarial Networks for Synthetic COVID-19 Ultrasound Data Generation

Pedro S. T. F. Silva
Department of Electrical Engineering
COPPE - UFRJ
Rio de Janeiro, Brazil
pedrosergiot@lps.ufrj.br

Antônio M. F. L. M. de Sá
Department of Biomedical Engineering
COPPE - UFRJ
Rio de Janeiro, Brazil
amflms@peb.ufrj.br

Leonardo B. Felix
Department of Electrical Engineering
UFV
Viçosa, Brazil
leobonato@ufv.br

Wagner C. A. Pereira
Department of Biomedical Engineering
COPPE - UFRJ
Rio de Janeiro, Brazil
wagner@peb.ufrj.br

José M. Seixas
Department of Electrical Engineering
COPPE - UFRJ
Rio de Janeiro, Brazil
seixas@lps.ufrj.br

***Abstract*—Lung ultrasound emerges as a powerful tool for the diagnosis of COVID-19, being a very cost-effective option to other modalities of exams, such as computerized tomography and X-ray imaging. There are efforts in trying to employ deep learning to develop systems that can make an automatic diagnosis based on ultrasound exams to assist the medical decision, but they are limited by the amount of data available. The present work tackles this problem by proposing a method using generative adversarial models to create synthetic data and increase the volume of data to train more complex models. To evaluate whether the synthetic data presents a variety close to that of the original data without replicating training samples, it was devised applications of the Kullback-Leiber divergence and L1 norm. Results indicate that the generated data sampled the main features of the ultrasound data, presenting a variety close to the original data. This points to the possibility of using the proposed method as a means to overcome the problem of low data volume for lung ultrasound. which is shown using expert knowledge.**

***Index Terms*—deep learning, generative adversarial networks, lung ultrasound, COVID-19**

I. Introduction

In 2019, COVID-19 was detected and the World Health Organization (WHO) declared a global pandemic in 2020, affecting millions of people around the world [1]. This led to an urgent need for research related to the disease diagnosis, with a great emphasis on the evaluation of the respiratory tract, since the lungs are the most affected organs by COVID-19 [2]. In this scenario, the application of medical imaging techniques, such as computed tomography (CT) and X-ray exams, presents a way of complementing the diagnostic process, being especially useful in triage situations and accelerating the proper treatment [3].

Given how recent the disease is and its similarities with other pulmonary disorders (such as other forms of pneumonia), the diagnosis performance through imaging can be greatly affected by the experience and expertise of the radiologists, also being a very time-consuming task [4]. This opens up an opportunity for the development of artificial intelligence (AI) systems for accurate image-based diagnosis in the context of COVID-19. This kind of approach (especially those employing deep learning models) has already been studied for the automated diagnosis of other diseases and conditions, such as the identification of malignant nodules in CT scans and detection of signs of tuberculosis in X-ray exams [5], [6].

Even though CT and X-ray are commonly adopted as the main options for screening lung diseases, lung ultrasound (LUS) is gaining more and more space due to having a good number of advantages such as being cheaper, non-invasive, repeatable, portable, and safer in the sense of not exposing the patient to non-ionizing radiation (which occurs in CT or X-ray exams) [3], [7], [8]. Because of that, this exam modality has drawn attention during the COVID-19 outbreak, with some protocols being designed to apply LUS to assess a patient's condition by means of the analysis and quantization of some relevant ultrasound findings [7]. More recently, some studies investigated the feasibility of applying deep-learning techniques for the automatic classification of conditions regarding the context of COVID-19 based on LUS exams: Born and colleagues [3] applied transfer learning from a pre-trained model both for ultrasound videos and images, achieving overall accuracies of 87% and 90%, also identifying the localization of biomarkers in the exams; Roberts and Tsiligkaridis [9] presented models with average accuracies ranging from 81% to 85%, also pointing the advantages of training robust models that try to minimize the effects of adversarial attacks; Baum et al [11] achieved accuracy values around 95% for the binary classification through the use of an image quality assessment module before the classification model; Awasthi et al. [12] proposed a light and efficient deep learning model for detection of COVID-19 (Mini-COVIDNet), reporting accuracy of 83,20%. However, there is still a lack of LUS data for COVID-19, with only a few collections available publicly, which poses an obstacle for the training of complex models [13].

In the present work, the possibility of applying generative models was investigated as a means of producing synthetic COVID-19 LUS, which could provide a handle for the low data volume available and enable the development of efficient deep learning models. This approach has already been studied for X-ray and CT in other studies ([14], [15], [16]), but has not yet been explored using LUS exams.

II. Method

A. Ultrasound Data

This study used a dataset publicly available [3], consisting of LUS exams from 216 patients diagnosed with COVID-19, bacterial pneumonia, viral pneumonia, and healthy. A summary of the dataset is given in Table 1, giving the number of exams for each class, the type of transducer used to collect them, and the format of the file (video or image). Since there were so few examples for the class viral pneumonia, we discarded it and worked with the remaining three conditions. Also, we used only data collected using a convex transducer for the following analysis.

TABLE I
Composition of the LUS dataset used. Adapted from [3].

	Convex		Linear		
Classe	Vid	Img.	Vid	Img.	Sum
COVID-19	64	18	6	4	**92**
Bact. Pneu.	49	20	2	2	**73**
Viral. Pneu	3	-	3	-	**6**
Healthy	66	15	9	-	**90**
Total	**182**	**53**	**20**	**6**	**261**

The processing of these data followed the same procedures used in [3], with frames being extracted from the videos (max 30 frames per video), cropping them to a quadratic window (excluding texts and border artifacts) and resizing the resultant images to 112 x 112 pixels (this size of images was used due to limitations of the hardware used).

B. Generative Models

For generating the synthetic data, GAN models are employed, which actually consist of two artificial neural networks (ANN) that are trained simultaneously using a competitive process [17]. The original GAN framework was proposed by Goodfellow and colleagues in [18], consisting of a zero-sum game where the two players are a generator network and a discriminator network. The generator $G(\mathbf{z})$ tries to learn the distribution of the original data through a continuous improvement of data mapping of a prior distribution $p_{\mathbf{z}}(\mathbf{z})$ and observing whether the resulting response would belong to such a distribution. Usually, $\mathbf{z}$ comes from a noise process (typically, white Gaussian). On the other side, the discriminator network $D(\mathbf{x})$ is trained to learn the mapping of a given input data $\mathbf{x}$ onto a scalar number that gives the probability of that input being an observation from the original dataset. During the adversarial training the model D learns how to maximize the probability of correctly classifying original and synthetic data, while G is trained to generate images that would not be detected as synthetic by D. The whole training can be summarized by the following optimization process:

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

where p_{data} is the probability distribution of the original data. Training goes until the discriminator cannot correctly identify which observations are synthetic and which are original data.

When compared to other methods previously developed, such as the Boltzmann Machine or Autoencoders, GANs present great advantages in terms of computational cost and fewer restrictions to the generator [19]. However, the training of GANs is sometimes unstable, with many studies publishing heuristics that can result in more stable architectures [20]. One of the major problems faced when training GANs is the mode collapse, which happens when the model can generate plausible synthetic observations but they do not cover all the diversity of the original data. The Wasserstein GAN (WGAN) [20] is a variation of the original GAN that tries to solve both the problems of stability and mode collapse.

As described by Arjovsky et al. [20], the WGAN changes the task executed by the discriminator as well as the objective function used during the training phase. The training for the WGAN can be seen as the minimization of the distance between the distribution of the original data and the distribution of the synthetic data. To achieve this, the objective function used is the Wasserstein-1 distance (or Earth-Mover distance) given by

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \prod(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}[\|x - y\|], \quad (2)$$

with $\prod(\mathbb{P}_r, \mathbb{P}_g)$ as the set of all joint distributions $\gamma(x, y)$ that have $\mathbb{P}_r$ and $\mathbb{P}_g$ as marginals [19]. The $\gamma(x, y)$ can be imagined as the mass that should be transported from x to y to transform the distribution $\mathbb{P}_g$ into $\mathbb{P}_r$, with $W(\mathbb{P}_r, \mathbb{P}_g)$ as a measure of the optimized energy cost for this transport. Since the infimum in 2 is intractable, it can be rewritten using the Kantorovich-Rubinstein duality as [21]

$$W(\mathbb{P}_r, \mathbb{P}_g) = \sup_{\|f\|_L \leq 1} \mathbb{E}_{x\sim\mathbb{P}_r}[f(x)] - \mathbb{E}_{x\sim\mathbb{P}_g}[f(x)], \quad (3)$$

being the supremum calculated with respect to all 1-Lipschitz functions. The supremum is still intractable but is easier to approximate, which enables an implementation of the Wasserstein distance.

Using 3, the problem boils down to finding the function f that maximizes the result of the equation. As described in [21], it can be considered an ANN with parameters w contained in a space $\mathcal{W}$ that obeys the Lipschitz restrictions for f. Considering all functions f_w that fit the desired criteria, it can be written

$$W(\mathbb{P}_r, \mathbb{P}_g) \approx \min_G \max_{w \in \mathcal{W}} \mathbb{E}_{x \sim \mathbb{P}_r}[f_w(x)] - \mathbb{E}_{z \sim p_{\mathbf{z}(\mathbf{z})}}[f_w(G(z))]. \quad (4)$$

However, to ensure that f_w follows the Lipschitz restriction, the authors impose a limitation to the weights of the neural network, clipping them to a range [-0.01, +0.01]. This approach has some drawbacks, such as limiting the capacity of the ANN and even generating some instability during the training. To circumvent these problems, it was employed the variation WGAN Gradient Penalty (WGAN-GP) presented in [22], which uses the following objective function during the models training:

$$L = \mathbb{E}_{\tilde{x} \sim \mathbb{P}_g}[D(\tilde{x})] - \mathbb{E}_{x \sim \mathbb{P}_r}[D(x)] + \lambda \mathbb{E}_{\hat{x} \sim \mathbb{P}_{\hat{x}}}[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2], \quad (5)$$

with $\hat{x}$ as interpolated observations between the original data and the synthetic data produced by the generator and λ as a penalty coefficient.

In the present work, WGAN-GP models were trained separately for each of the three classes (COVID-19, bacterial pneumonia, and healthy), building class-expert generative models. Fig. 1 presents the architecture for these models. The discriminator (or critic) was a convolutional neural network (CNN) composed of three convolutional layers (128 filters each, 3x3 kernels, and stride equal to 2), a fully connected layer with 128 units, and an output layer with a single unit. The activation function used in all layers was rectified linear unit (ReLU), except for the output layer which used a linear activation function.

The generator network consisted of a fully connected layer with 25088 units, three transpose convolution layers (each with 128 filters, 3x3 kernels, and stride equal to 2), and an output layer that is also a transpose convolution layer (1 filter, kernel size equal to 3x3 and stride equal to 1). The input for this model was a vector of 100 numbers which came from a spherical Gaussian distribution. The hyperbolic tangent was the activation function for the output layer, while all the other layers used ReLU. Batch normalization was applied to the output of each layer except the output layer (it was not used in the discriminator network since it could modify the loss function used in the WGAN-GP, as pointed out in [22]).

The WGAN-GP models were trained for 20,000 epochs, with the discriminator being updated 5 times for each update of the generator. The batch size was equal to 64 images, and both the discriminator and generator used the optimizer Adam (learning rate = 0.0001, $\beta 1$ = 0.5 and $\beta 2$ = 0.999). The training for each model followed the cross-validation k-fold with k=10, which means that 10 models were developed for each class using different partitions for training and test sets [23].

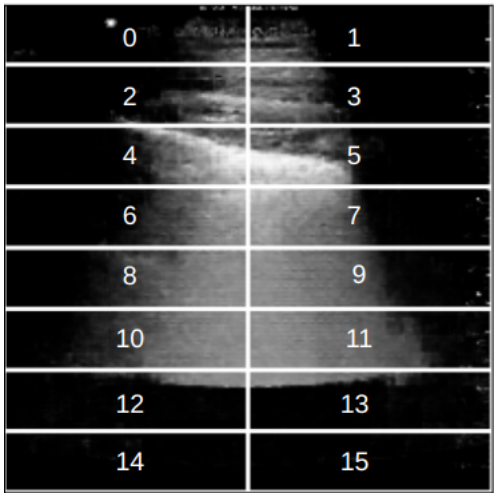

Fig. 2. Example of an image from the original dataset divided into the 16 patches used.

C. Performance Measures

There is still much discussion about how to measure the variety and quality of the synthetic data [24]. In the present work, we used a similar method as the one presented in [25], which employs the Kullback-Leiber (KL) divergence as a form of comparing the variation of the synthetic data is contained within the variation of the original data. We used the following procedure:

- each synthetic or original image was divided it into 16 patches, as shown in the Fig. 2;
- each pair of original images in the training set was compared at patch level using the KL divergence to check how far the distribution of pixels of a patch in one of the images is from the other one;
- the same procedure was applied to compare each pair consisting of one original image from the training set and one synthetic image, also at patch level;
- The two sets of quasi-distance values obtained for each patch are compared to check whether the generated images present a variance close to that estimated in the original data.

The division of the images into patches was done in order to compare specific regions of the generated images to the original ones. This approach was chosen (instead of comparing the whole images) due to the localization of important LUS findings used in the medical diagnosis. To that end, the size of the patch was defined such that the pleural line (one of the most important findings for the diagnosis) of most of the images was contained in one or two subsequent patches. Through the comparison of the KL divergence values for the original-original combinations and the original-synthetic combinations, it is possible to evaluate whether the difference fluctuations between the synthetic images and original images are within the bounds of that estimated among the original data. That would be an indication that the generated data follows the same probability density function as the original

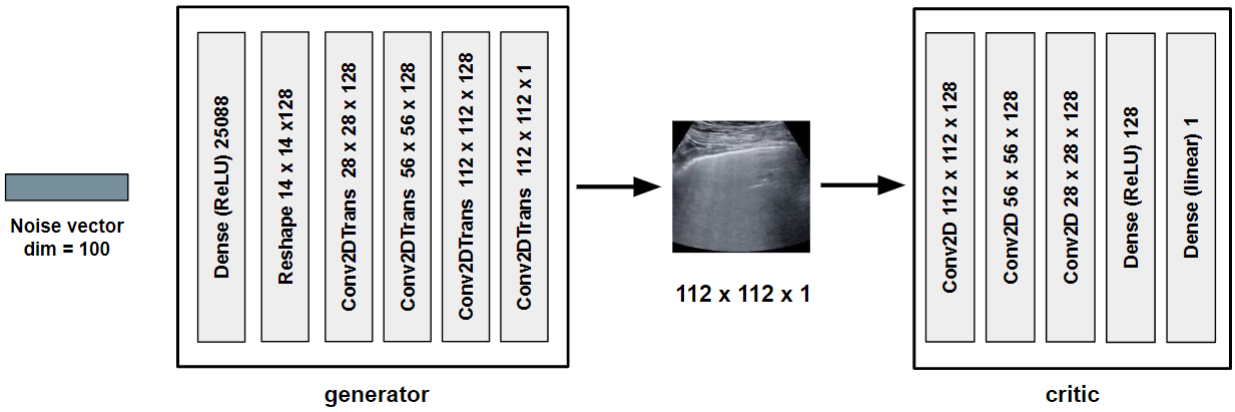

Fig. 1. Architecture of the WGAN-GP model implemented for the synthetic data generation.

data. And since we divide the image into patches, it can be checked if the generative model is failing for specific regions of the LUS.

Although the KL divergence can be used to compare the variances and check the generated data adherence to the pdf, it does not point out whether the generator is only replicating data in the training set. To that end, it was employed the calculation of the l1 norm, as described in [25]. By comparing the histograms of this to measure estimated for each pair of original-original images and original-synthetic, it was possible to see if the generator is only creating copies of the training data (which would result in some entries equal to 0 in the histogram for original-synthetic norms) and if the two histograms are close to each other but still presenting some differences (meaning the produced data has a distribution close to the training data but can also expand this distribution).

III. Results and Discussion

The training curves for the models of the first fold of the k-fold cross-validation are shown in the first column of Fig. 3. For each model, there are two curves: the training loss (Wasserstein distance) and a filtered version of this loss using a moving average filter with a window size of 50 epochs. The filtered curve is shown to check if the loss already converged or whether it keeps decaying since other studies cited this convergence as an indication of a successful WGAN training [20], [22], [25]. Although there are differences between the three curves, there is a similar behavior: a quick decay at the beginning of the training followed by an increase in the training loss and a slow decay until the max number of epochs. As the critic was beginning to learn how to differentiate synthetic from original data in the first epochs, it may have had some difficulty when performing this task, which could be the cause for the first oscillation in the related behavior. Also, The Wasserstein distance for the COVID-19 and Pneumonia classes (A) and B) in Fig. 3) still show signs of decay after the 20000 training epochs, which means that there is still some room for improvement of these models by using more training epochs. A similar behavior was observed in the training curves for the remaining cross-validation folds.

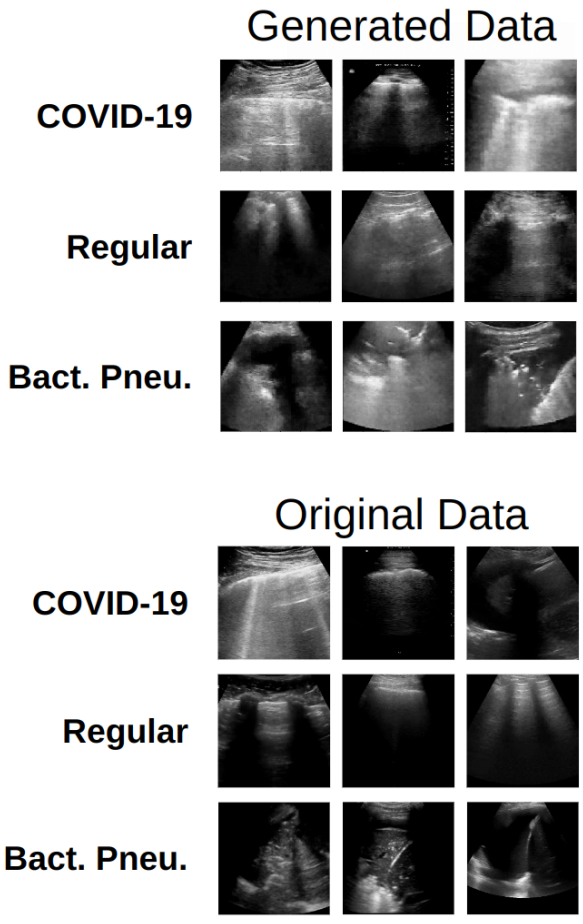

Fig. 4. Examples of images generated and original images for each of the classes.

Next, 5000 images for each class were generated using the trained generators. Some of the synthetic images for fold 1 are shown in Fig. 4 along with images from the original

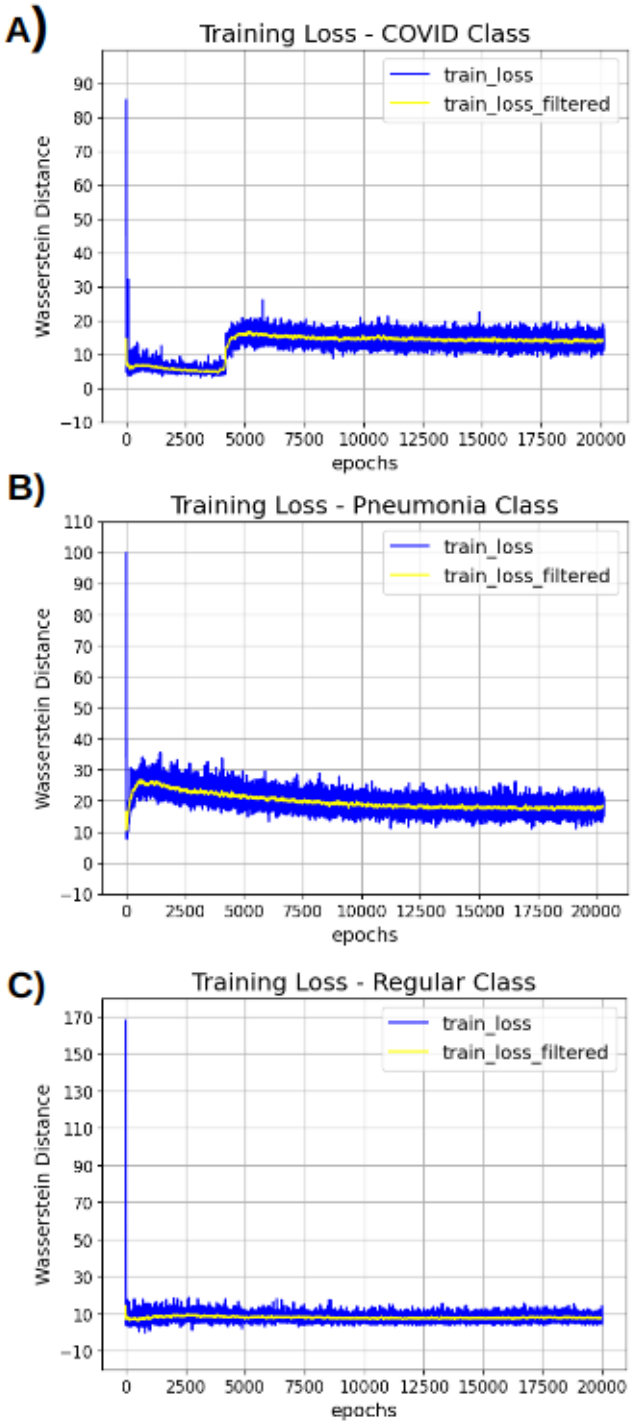

Fig. 3. Training curves for each of the three expert WGAN-GPs trained in fold 1: A) COVID-19 class, B) Pneumonia class and C) Regular class.

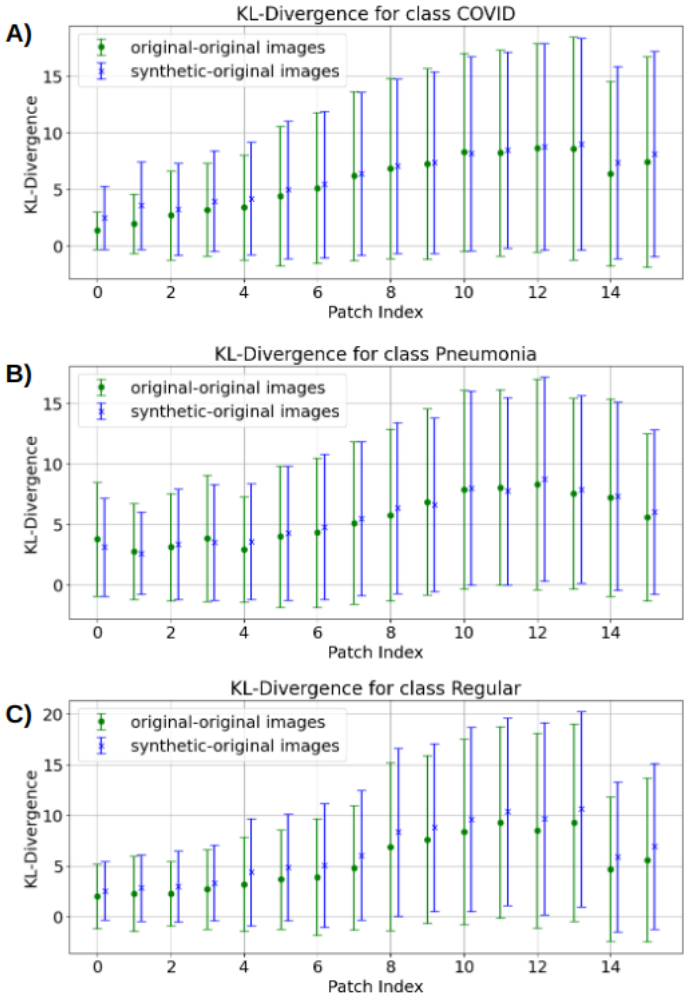

Fig. 5. KL divergence estimated for each pair of original images and each pair synthetic-original, at patch level, for fold 1 of the three classes

dataset. It is possible to see that the synthetic/generated images are very similar to the original data, even showing some important findings used in medical diagnostics, such as the pleural line, coalescent B-lines, and signs of consolidation for the bacterial pneumonia class. This is especially relevant for synthetic data production, since these findings are essential for the identification of the different conditions, with LUS medical diagnosis protocols being based on their presence and quantization [7].

The evaluations using the KL divergence and l1 norm were then employed to check whether the synthetic data also presented a variety close to that of the original data and whether or not the model was only replicating examples from the training set. Fig. 5 shows this result for fold 1 of the k-fold cross-validation (results were almost the same for the remaining folds). It can be noted that the variability of the KL divergence for each patch in the original data is close to the variety of KL divergences estimated using pairs of original-synthetic data, which is a good indicator that the models are able to generate the different modes present in the original

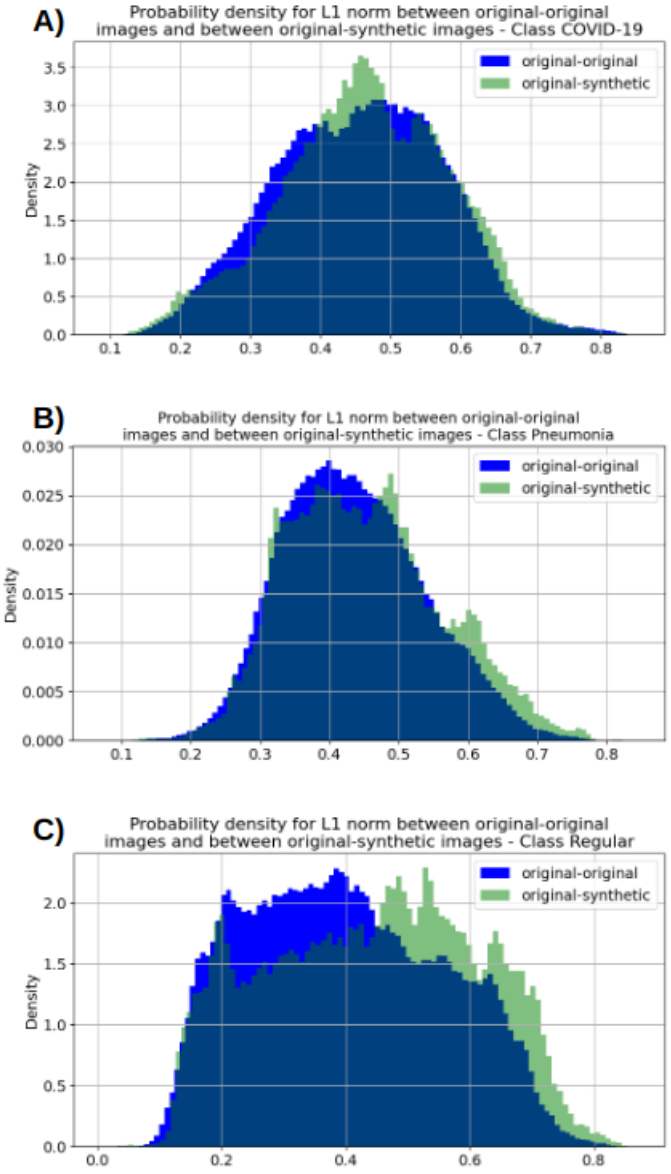

Fig. 6. L1 norm between original-original and original-synthetic images for classes A) COVID-19, B) Bacterial Pneumonia and C) Regular.

data.

The results obtained by applying the l1 norm in fold 1 are shown in Fig. 6, which also occurred repeatedly in all the other folds. Since there are no distribution entries equal to zero, the generators of each class are not replicating any of the training examples. Also, the distributions for the original-original and for the original-synthetic pairs are very close but still present some small differences (especially for the class Regular). This indicates a general similarity between the generated data and the training set, which is coherent with the examples shown in Fig. 4.

Summarizing, the results for both measures together with the presence of the main features used for medical diagnosis in the generated data, points to the possibility of employing the proposed method as a means of overcoming the scarcity of data for training COVID-19 LUS-based models.

IV. Conclusion

This study presented a method for generating artificial LUS data in the context of the disease COVID-19, tackling the problem of data scarcity for this type of exam. To achieve this goal, WGAN-GP models were trained in order to generate ultrasound images that followed the distribution of a given sample.

The generated data was very similar to the original examples, presenting the main ultrasound findings related to each of the three classes present in the dataset (COVID-19, Bacterial Pneumonia, and Regular). To check if the synthetic data really followed the distribution close to that of the original data, presenting a similar variety and not replicating the training data, an application involving the KL divergence and l1 norm was employed. The results demonstrate that the variance of KL divergence, when comparing generated and original images, closely mirrors that estimated among the original data. The l1 norm analysis further indicates that the generated data is not mere replication of training images. This is observed across different image regions and the dataset's three classes (bacterial pneumonia, COVID-19, and healthy), reinforcing the notion that the generated data faithfully follows the original data's distribution without replication.

The outcomes underscore the potential of the proposed method in circumventing the data scarcity issue in lung ultrasound studies. This approach opens avenues for experimentation with more sophisticated models to enhance automatic patient diagnosis using deep learning methods.

Future work will investigate whether synthetic images can be used to improve the performance of LUS-based classification methods.

Acknowledgment

The authors thank CNPq and FAPERJ for their support to this study. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) Finance Code 001.

References

[1] M. Yuce, E. Filiztekin, K. G. Ozkaya, "COVID-19 diagnosis - A review of current methods," Biosensors and Bioelectronics, vol. 172, pp. 112752, 2021.

[2] F. Guarraciono, et al., "Lung, heart, vascular, and diaphragm ultrasound examination of COVID-19 patients: A comprehensive approach, " Journal of Cardiothoracic and Vascular Anesthesia, vol. 35, pp. 1866–1874, 2021.

[3] J. Born, et al., "Accelerating detecntion of lung pathologies with explainable ultrasound image analysis, " Applied Sciences, vol. 11, num. 2, pp. 672, 2021.

[4] P. Aggarwal, et al., "COVID-19 image classification using deep learning: Advances, challenges and opportunities," Computers in Biology and Medicine, vol. 144, pp. 105350, 2022.

[5] T. Meraj, et al., "Lung nodules detection using semantic segmentation and classification with optimal features," Neural Computing and Applications, vol. 33, pp. 10737–10750, 2021.

[6] A. T. Sahlol, M. Abd Elaziz, A. Tariq Jamal, R. Damasevicius and O. Farouk Hassan, "A novel method for detection of tuberculosis in chest radiographs using artificial ecosystem-based optimisation of deep neural network features," Symmetry, vol. 12, num. 7, pp. 1146, 2020.

[7] G. Soldati, A. Smargiassi and R. Inchingolo, "Proposal for international standardization of the use of lung ultrasound for patients with COVID-19: A simple, quantitative, reproducible method," Journal of Ultrasound in Medicine, vol. 39, num. 7, pp. 1413–1419, 2020.
[8] L. Zhao and M. A. L. Bell, "A review of deep learning applications in lung ultrasound imaging of COVID-19 patients," BME Frontiers, vol. 2022, 2022.
[9] J. Roberts and T. Tsiligkaridis, "Ultrasound diagnosis of COVID-19: Robustness and explainability," arXiv preprint arXiv:2012.01145, 2021.
[10] Z. Baum, et al., "Image quality assessment for closed-loop computer-assisted lung ultrasound," Medical Imaging 2021: Image-Guided Procedures, Robotic Interventions and Modeling, vol. 11598, 2021.
[11] J. Diaz-Escobar, et al., "Deep-learning based detection of COVID-19 using lung ultrasound imagery," PLoS ONE, vol. 16, num. 8, 2021.
[12] N. Awasthi, A. Dayal, L. R. Cenkeramaddi and P. K. Yalavarthy, "Mini-COVIDNet: Efficient lightweight deep neural network for ultrasound based point-of-care detection of COVID-19," IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 68, num. 6, pp. 2023-2037, 2021.
[13] J. Wang, et al., "Review of Machine Learning in Lung Ultrasound in COVID-19 Pandemic," Journal of Imaging, vol. 8, num. 3, pp. 65, 2022.
[14] A. Sedik, et al., "Deploying machine and deep learning models for efficient data-augmented detection of COVID-19 infections," Viruses, vol. 12, num. 7, pp. 769, 2020.
[15] U. Kiru, et al., "Comparative analysis of some selected generative adversarial network models for image augmentation: a case study of COVID-19 x-ray and CT images," Journal of Intelligent and Fuzzy Systems, vol. 43, num. 6, pp. 7153-7172, 2022.
[16] P. M. Shah, et al., "DC-GAN-based synthetic X-ray images augmentation for increasing the performance of EfficientNet for COVID-19 detection," Expert Systems, vol. 39, num. 3, 2022.
[17] A. Creswell, et al., "Generative adversarial networks: An overview," IEEE Signal Processing Magazine, vol. 35, num. 1, pp. 53–65, 2018.
[18] I. Goodfellow, et al., "Generative adversarial networks," Communications of the ACM, vol. 63, num. 11, pp. 139–144, 2020.
[19] H. Alqatani, M. Kvakli-Thorne and G. Kumar, "Applications of generative adversarial networks (GANs): An updated review," Archives of Computational Methods in Engineering, vol. 28, pp. 525–552, 2021.
[20] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," arXiv preprint arXiv:1701.04862, 2017.
[21] M. Arjovsky, M. Chintalas and L. Bottou, "Wasserstein generative adversarial networks ," International conference on machine learning, pp. 214–224, PMLR, 2017.
[22] I. Gulrajani, F. Ahmed and M. Arjovsky, "Improved training of Wasserstein GANs," Advances in neural information processing systems, vol. 30, 2017.
[23] M. Artur, "Review the performance of the Bernoulli Naïve Bayes Classifier in Intrusion Detection Systems using Recursive Feature Elimination with Cross-validated selection of the best number of features, " Procedia Computer Science, vol. 190, pp. 564–570, 2021.
[24] A. Borji, "Pros and cons of GAN evaluation measures," CoRR, vol. abs/1802.03446, 2018, available at: http://arxiv.org/abs/1802.03446
[25] J. C. V. Fernandes, N. N. Moura Junior, and J. M. Seixas, "Deep learning models for passive sonar signal classification of military data," Remote Sensing, vol. 14, pp. 2648, 2022.