Learning and Nonlinear Models - Journal of the Brazilian Society on Computational Intelligence (SBIC), Vol. 18, Iss. 2, pp. 4-15, 2020

COMPARATIVE ANALYSIS OF CONVOLUTIONAL NEURAL NETWORKS APPLIED IN THE DETECTION OF PNEUMONIA THROUGH X-RAY IMAGES OF CHILDREN

Luan Oliveira da Silva
Universidade Federal do Sul e Sudeste do Pará
luansilvatec@unifesspa.edu.br

Leandro dos Santos Araújo
Universidade Federal do Sul e Sudeste do Pará
leandronetlink@unifesspa.edu.br

Victor Ferreira Souza
Universidade Federal do Sul e Sudeste do Pará
victoor.soouza@unifesspa.edu.br

Raimundo Matos de Barros Neto
Universidade Federal do Sul e Sudeste do Pará
netobarros@unifesspa.edu.br

Adam Santos
Universidade Federal do Sul e Sudeste do Pará
adamdreyton@unifesspa.edu.br

Abstract – Pneumonia is one of the most common medical problems in clinical practice and is the leading fatal infectious disease worldwide. According to the World Health Organization, pneumonia kills about 2 million children under the age of 5 and is constantly estimated to be the leading cause of infant mortality, killing more children than AIDS, malaria, and measles combined. A key element in the diagnosis is radiographic data, as chest x-rays are routinely obtained as a standard of care and can aid to differentiate the types of pneumonia. However, a rapid radiological interpretation of images is not always available, particularly in places with few resources, where childhood pneumonia has the highest incidence and mortality rates. As an alternative, the application of deep learning techniques for the classification of medical images has grown considerably in recent years. This study presents five implementations of convolutional neural networks (CNNs): ResNet50, VGG-16, InceptionV3, InceptionResNetV2, and ResNeXt50. To support the diagnosis of the disease, these CNNs were applied to solve the classification problem of medical radiographs from people with pneumonia. InceptionResNetV2 obtained the best recall and precision results for the Normal and Pneumonia classes, 93.95% and 97.52% respectively. ResNeXt50 achieved the best precision and f1-score results for the Normal class (94.62% and 94.25% respectively) and the recall and f1-score results for the Pneumonia class (97.80% and 97.65%, respectively).

Keywords – Deep Learning, Pattern Recognition, Convolutional Neural Networks, Pneumonia.

1. INTRODUCTION

Object recognition in images is a simple activity for humans and is becoming increasingly easy for computers due to advances in computer vision studies using deep learning neural networks. Deep learning is a sub-area of machine learning, which employs algorithms to process data, drawing on the processing carried out by the human brain, using a set of connected neurons that are arranged in layers such that each one processes the information and passes the obtained result from a given input to the next layer [1].

These algorithms, especially the convolution-based ones, have made great strides since 2012 and have been used in a variety of classification problems involving general images, even medical images. The application of deep learning techniques for medical image classification has grown considerably in recent years. Many studies addressed this issue, such as performing image classification to aid in the early diagnosis of tuberculosis, and classifying lesions by means of chest radiography [2]. Besides, some works also applied deep learning techniques for classifying x-ray images of people with pneumonia and other

diseases [3, 4]. Specifically, pneumonia is a form of acute respiratory infection that affects the lungs. These respiratory organs are made up of small sacs called alveoli, which fill with air when a healthy person breathes. When an individual has pneumonia, the alveoli become full of pus and fluid, which makes breathing painful and limits oxygen intake [5].

This research presents five implementations of convolutional neural networks (CNNs): ResNet50, VGG-16, InceptionV3, ResNeXt50, and InceptionResNetV2. They have different paradigms, especially in relation to their architecture, for the extraction of image characteristics. Therefore, they are feasible choices for comparative studies in this area. These CNNs are applied to solve the classification problem of x-ray medical image from people with pneumonia to aid the disease diagnosis. The goal of this work is to make a comparative study between the implementations, employing a regularization-based methodology (via Dropout and Data Augmentation) and a dynamic learning rate during the training phase. Afterwards, the best performance in the image classification is evaluated. This paper is structured as follows. Section 2 presents the motivation of this research. Section 3 highlights the related works. Section 4 covers the concept of CNN, its structure and arrangement of layers. In addition, section 4 also encompasses the architectures that were used. Section 5 comprises the methodology applied in this work, as well as the database that was used. Section 6 discusses the results from the implementations of the five architectures. Finally, section 7 includes the final considerations and possible future works to be developed from this study.

2 MOTIVATION

According to the World Health Organization (WHO), pneumonia kills about 2 million children under the age of 5 each year and is constantly considered as the leading cause of child mortality, killing more children than AIDS, malaria, and measles combined [5].

The WHO reports that almost all cases of children with pneumonia occur in developing countries, particularly in Southeast Asia and Africa. Bacterial and viral pathogens are the two main causes of pneumonia, but they require very different forms of management. Bacterial pneumonia requires urgent referral for immediate antibiotic treatment, while viral pneumonia is treated with caution. Thus, accurate and timely diagnosis is indispensable. A key element in the diagnosis is radiographic data, as chest radiographs are routinely obtained as a standard of care and can aid differentiate between types of pneumonia [3]. However, rapid radiological interpretation of images is not always available, particularly in places with minor resources, where childhood pneumonia has the highest incidence and mortality rates.

The process of interpreting agents (brain tumors or lung anomalies, for example) is a complex activity, so it is necessary to use image processing techniques, often combined with machine learning techniques, to identify them [6]. The analysis and diagnosis of chest radiographs play a very critical role due to their availability and accessibility, as a specialist radiologist is required to interpret the results of a medical exam.

A computer-assisted sorting system would mitigate these issues by providing the initial interpretation of Chest X-Ray (CXR) to extract a normal/infected image classification. The Computer-Aided Diagnosis (CAD) concept for chest radiographs has been around for fifty years. Traditionally, CAD systems use the machine learning technique to perform data analysis tasks [2].

3 RELATED WORK

In [7], an advanced system for the classification of pneumonia and tuberculosis through x-ray images was developed, namely Artificial Intelligence Aided Screening (AIAS). It was proposed an algorithm based on the Inception architecture, with 211 layers and 45 million trainable parameters. It was used four different databases, which are: ChestXray14, OCT, and Chest X-Ray from Mendeley Data, M-SC-Xray, and Belarus, respectively. The metrics for evaluating the results were area under the curve (AUC), Sensitivity or true positive rate (TPR), and Specificity. It was applied the transfer learning paradigm in the training stage, generating a pre-trained model from the ChestXray-14 database, seeking to make the model as general as possible. In addition, data augmentation was applied to generate new images, applying changes in scale, rotation, translation, and so on. An AIAS system was formulated as a multi-label classification problem, in which each class is independent and not mutually exclusive, i.e., a single x-ray image can contain the pathological characteristics of Tuberculosis and Pneumonia at the same time. The authors demonstrated the results according to the metrics and compared them with other related works. A limitation of this work is that the model was trained using frontal images of the chest, which yielded the algorithm's inability to accurately diagnose radiographs with lateral views.

In [8], it was developed a deep learning architecture based on ResNet (with 51 layers) that used dilated convolution to detect childhood pneumonia. This extended convolution was employed to ensure that the original resolution of the incoming network remained, thus decreasing the loss of resolution. Specifically, based on an understanding about the nature of the pneumonia image classification task, the method used the residual structure to overcome the problems of overfitting and degradation of the depth model. In addition, to overcome the problem of model training difficulty due to insufficient data, model parameters learned in large-scale data sets were used, initializing the model through transfer learning using pre-trained weights from the database. This method was evaluated to extract texture characteristics associated with pneumonia and to identify the performance of the image areas that best indicate pneumonia. A comparative analysis between the proposed architecture and VGG16, DenseNet121, Xception, and InceptionV3, was carried out. The experimental results on the test data showed that the proposed architecture achieved recall rate as 96.7%, accuracy as 90.5%, AUC as 95.3% and f1-score as 92.7%, being surpassed in the precision metric by InceptionV3 (89.1% against 91.6%). Therefore, it performed better than classic architectures. Compared with the prior state-of-the-art methods, this approach can effectively solve the problem of low image resolution and partial occlusion of the inflammatory area in chest x-ray images.

In [9], it was proposed a trained CNN model from scratch to classify and detect the presence of pneumonia from a collection of chest x-ray samples. The authors pointed out that to avoid overfitting and make the model generalizable, data augmentation was applied. This model consists of two parts, with a total of 12 layers: the feature extractor and the classifier. The validation of the model was carried out through the application of the algorithm in databases of different resolutions, with accuracy as the standard metric. It was performed 10 simulations. The model obtained an average accuracy higher than 94% in training and 93% in validation. The authors emphasized that in their study they built the model from scratch, in contrast to other methods that depend heavily on the transfer learning approach.

Table1 synthesizes the correlation between the literature works and the proposed study. It is important to highlight that all works applied data augmentation techniques.

Table 1: Comparison between proposed and related works.

Work	Type of nets	Dataset	Metrics	Optimizer	Problem type
7.	Own implementation based on Inception with 211 layers.	ChestXray14, OCT and Chest X-Ray (Mendeley), M-SC-Xray e Belarus.	AUC, ROC, sensitivity and specificity.	Nesterov.	Multiclass classification and multi-label.
8.	Own implementation based on ResNet 51 layers with pre trained weights.	OCT and Chest X-Ray (Mendeley)	Confusion Matrix, recall, f1-score, precision, Accuracy, ROC e AUC.	Adam	Binary classification.
9.	Own implementation.	OCT and Chest X-Ray (Mendeley)	Arithmetic mean of the accuracy of the network application with datasets of 5 different resolutions.	Not specified.	Binary classification.
Proposed work	Networks without pre-training: VGG-16, InceptionResNetV2, ResNeXt50 InceptionV3 and ResNet50.	OCT and Chest X-Ray (Mendeley)	Accuracy, recall, precision, Confusion Matrix and f1-score.	Adam.	Binary classification.

4 CONVOLUTION NEURAL NETWORKS

CNN are biologically inspired architectures capable of being trained and learning/generalizing invariant representations of scale, translation, rotation, and related transformations [10, 11]. CNNs are one of the algorithms in the area known as deep learning and are designed for multi-dimensional data, making them feasible candidates for solving image recognition problems [12]. Similar to traditional computer vision processes, a CNN is able to apply filters to unstructured data while maintaining the neighborhood relationship between image pixels throughout network processing. Figure 1 demonstrates the structure of a CNN.

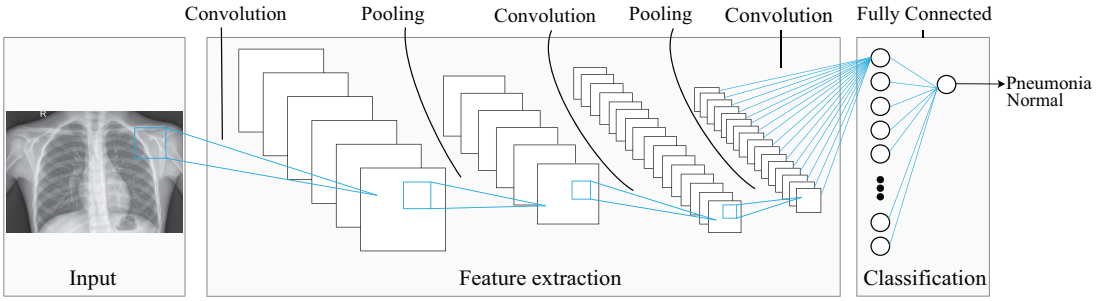


Figure 1: Structure of a CNN.

CNNs have been used by the scientific community in medical imaging because of their excellent performance in computer vision. Usually, three types of layers are used to build a CNN: convolutional, pooling, and fully connected layers [13].

Convolutional layers apply a number of filters over the image to obtain a series of feature maps, one for each filter. In turn, the grouping layers summarizes these feature maps highlighting the most relevant features. The classification task is performed by the fully connected layer [13].

The main layer of these networks is the convolution layer, and its function is to apply masks to the input images, based on a neighborhood of pixels. The output in this operation is the convolution filters (arrays) that store the weights of the connections between neurons [14, 15]. Convolution is represented by a linear operator between two functions that produces a third function. The latter represents the overlapping areas of the masks that are applied to the image. The distinction of data patterns with this operation is given by local receptive fields (small spatial portions of pixels), which are represented by structures of locally connected neurons [16].

The sharing weights on the convolution layer ensures that filters are applied at different positions on the image, significantly decreasing the number of data to be learned [17]. Activation is a very important structure in the convolution layer, making a fit between a set of neurons, with an activation function over a convolution. This produces feature maps that store the information learned by the filters [15].

Another layer that is also crucial in CNNs is the pooling layer. It reduces the dimensionality of feature maps by decreasing width and height. The pooling operation enables a spatial invariance. Feature pooling in most convolution architectures employs a Max pooling function, determining the maximum value of a group in a given neighborhood [14, 18].

The next layers of CNNs play the regression role of activations. In a given network, after the grouping layer, at least one fully connected (FC) layer is required, serving to create decision paths from the filters obtained in the previous layer [19]. The last layer of CNNs is also fully connected and can classify the data. In this situation, an activation function determines the identification of the outputs in classes. The most commonly used activation is the Softmax function for multi-class problems and Sigmoid for binary problems. Softmax determines the probability that an example belongs to a class according to its occurrence [15].

The training of a CNN is performed, in most cases, through the backpropagation algorithm, which adjusts the neuron weights by propagating and correcting errors related to data classification, supported by the optimization provided by the gradient descent [20]. CNN architectures are designed to fit a range of images and classes, providing more robust pattern recognition. Many studies investigate the size of these architectures and their effectiveness. In this sense, the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) assesses identification and classification approaches that apply deep neural networks. One of its motivations is to promote architectures that make great progress [14]. In this study, five of these widely used CNN architectures are considered [2, 4]: ResNet50, VGG-16, InceptionV3, ResNeXt50, and InceptionResNetV2.

4.1 VGG-16

VGG-16 is a CNN model proposed by K. Simonyan and A. Zisserman [18]. It was one of the famous models submitted to ILSVRC-2014. It has improved AlexNet by replacing large convolutional filters (11 and 5 in the first and second convolutional layers, respectively) with multiple 3x3 size filters one after the other.

The architecture of the VGG-16 is shown in Figure 2. The input to the first convolution layer has the default size of 224x224, accepting other sizes. The image is passed through a stack of convolutional layers, where the filters were used with a very small 3x3 receptive field. Spatial pooling is performed by five layers considering a maximum function. This grouping is performed in a 2x2 pixel window [18].

Three FC layers follow a stack of convolutional layers (which have a different depth in different architectures): the first two have 4096 neurons each, the third performs the classification. In the final layer the Softmax function is applied. All hidden layers encompass the ReLU nonlinearity.

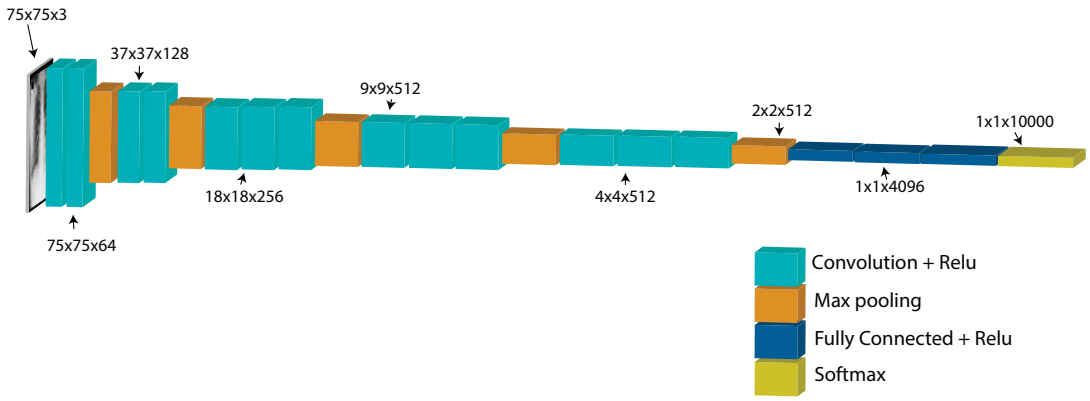


Figure 2: Architecture of the VGG-16.

4.2 ResNet50

ResNet or residual network is a classic CNN model used as a backbone for many computer vision tasks. This model was the winner of the 2015 ImageNet challenge. ResNet50 is a 50-layer residual network, but advances in its architecture allow one to train extremely deep neural networks with more than 150 layers. The ResNet50 architecture has two main features:

- "Identity shortcut connections": A strategy of "shortcuts" or "hop connections" that skip pairs of convolutional layer groups. They are also called gated units or gated recurrent units, although they do not recur in the traditional sense of recurrent neural network models;
- Heavy focus on batch normalization.

In a simple manner, the idea of shortcuts in ResNet50 is to prevent the very deep network from dying out by vanishing gradients through stacking identity mappings. Thus, as shown in Figure 3, ResNet50 uses at a given point a signal which is the sum of the signal produced by the two previous convolutional layers plus the signal transmitted directly from the point prior to these layers, by joining a processed signal with a signal from an earlier step [21].

The basic architecture of any ResNet is described in Figure 4. A zero padding block of (3,3) is applied to the input. In Stage 1, 2D Convolution has 64 shape filters (7,7) and uses a stride of (2,2). BatchNorm is applied to the input channel axis. Max

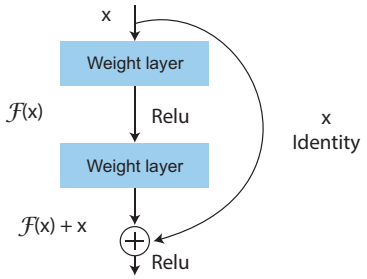


Figure 3: Residual Block.

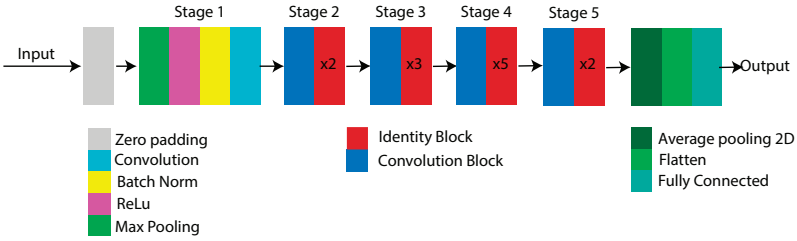


Figure 4: Architecture of the ResNet.

Pooling uses one window (3,3) and one pass (2,2). In Stage 2, convolutional block uses three sets of filters. The second identity block applies three sets of filters. In Stage 3, convolutional block employs three sets of filters. The third identity block uses three sets of filters. In Stage 4, the convolutional block uses three sets of filters. The fifth identity block includes three sets of filters. In Stage 5, convolutional block uses three sets of filters. The second identity block comprises three sets of filters. 2D Average Pooling uses a shape window (2,2). Flatten has no hyperparameter, and the FC layer reduces its input to the number of classes using a Softmax activation [21, 22].

4.3 InceptionV3

The GoogLeNet Network won the ILSVRC in 2014 [23]. Its main contribution was the development of an Inception module that drastically reduced the number of parameters in the network to 4 million, compared to the 60 million from AlexNet. The main purpose of the Inception module is to act as a feature extractor on many levels, computing 1x1, 3x3, and 5x5 convolutions within the same network module, as shown in Figure 5.

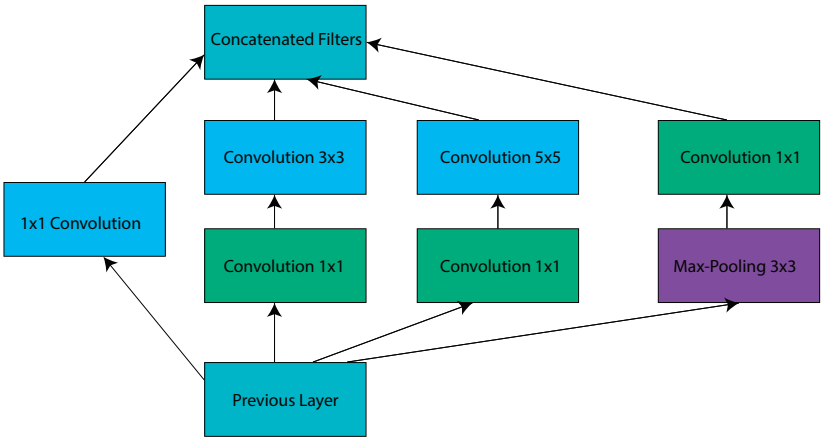


Figure 5: Inception Module.

InceptionV3 is a CNN model for addressing the image classification problem encountered in the ImageNet dataset. This network reduced the number of parameters, allowing a superior computational performance compared to VGG networks [23, 24]. As shown in Figure 6, the model is composed of symmetrical and asymmetrical components, including convolutions, Average Pooling (AvgPool), Max Pooling (MaxPool), Concatenations (Concat), Droupout, and fully connected layers. Batch normalization is used extensively throughout the model and applied to activation inputs. Classification is performed on the last layer using Softmax [25].

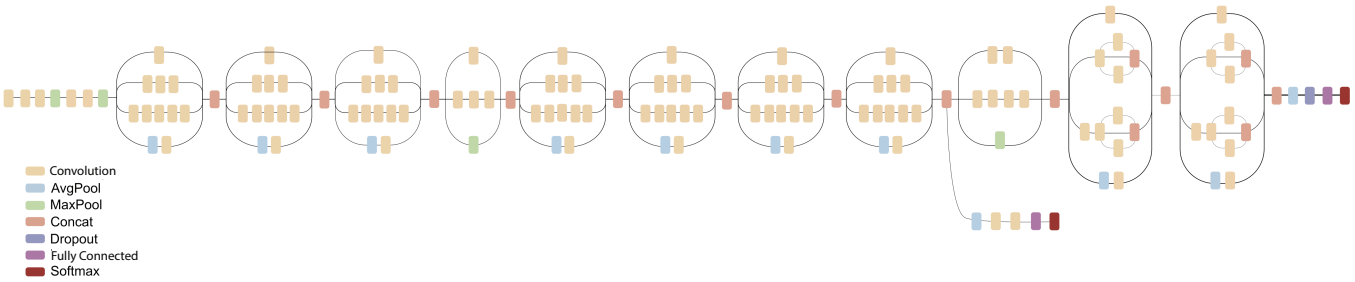


Figure 6: Architecture of the InceptionV3.

4.4 Inception-ResNetV2

The emergence of Inception-ResNet was due to the promising benefits this combination could bring. Inception-ResNet is basically the replacement of the concatenation of filters used in Inception by residual connections, as highlighted in Figure 7.

As Inception networks tend to be very deep, it is natural to replace the concatenation stage of this architecture with residual connections. This allows Inception to gather all the benefits from the residual approach, maintaining its computational efficiency [26].

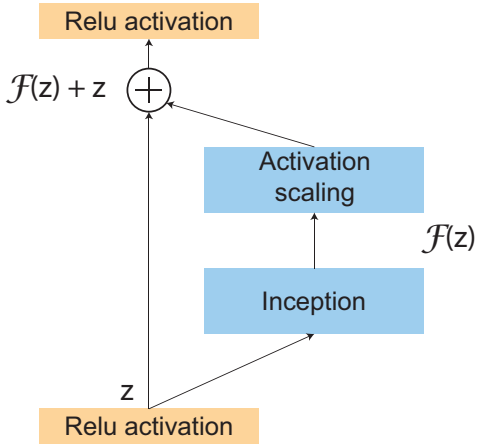


Figure 7: Inception-ResNet Module.

Inception-ResNet-v2 is a more expensive version of Inception with improved recognition performance.

4.5 ResNetXt

ResNetXt obtained the second place in the ILSVRC 2016 classification challenge [27]. In its architecture, the strategy of repeating layers and the use of residual blocks are employed, as well as the adoption of a module that explores the strategy of division, transformation, and merging. The main highlight of this network is the concept of cardinalities, which is the size of the module to carry out the transformations. The experiments carried out in [27] determine that increasing the depth and width of the network are less efficient than the increase in cardinality, to obtain a greater accuracy.

In Figure 8, it is possible to observe the transformation module. One of the highlights is the use of the same topology in the paths, which allows one to identify a greater number of transformations without the need to specialize the architecture. The cardinality value in the module is 32, that is, it has 32 equal paths.

5 METHODOLOGY

The programming language used for the implementation of CNN architectures was Python, along with Keras and TensorFlow libraries. Keras is a library that provides highly powerful and abstract building blocks for establishing deep learning networks [28, 29]. TensorFlow is a machine learning system that operates on a large scale and in heterogeneous environments [30].

The infrastructure used to carry out the simulations was Google Colab (Collaboratory), which offers 12GB of RAM and a Tesla K80 GPU. It provides pre-configured Python 2 and 3 runtimes with essential machine learning and artificial intelligence libraries, such as TensorFlow, Matplotlib, and Keras [31, 32].

Before addressing the experimental results, an analysis on the database, confusion matrix, data augmentation, and how the parameters for the tests were defined is carried out. For a more precise analysis, ten simulations were performed for each network, with the goal of obtaining an average behavior of each architecture.

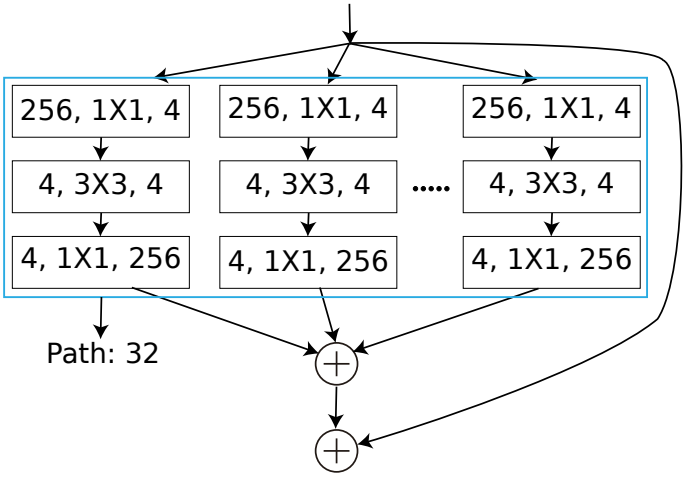


Figure 8: ResNeXt Module.

5.1 Database

For this research, a database with x-ray images of the chest from patients with different types of pneumonia was used, such as the examples in Figure 9 [3]. There are three classes: normal, viral pneumonia, and bacterial pneumonia. The images were collected and tagged from a total of 5,856 x-ray images from children's chest, including 4,273 categorized as representing pneumonia (2,779 bacterial and 1,494 viral) and 1,583 normal [3].

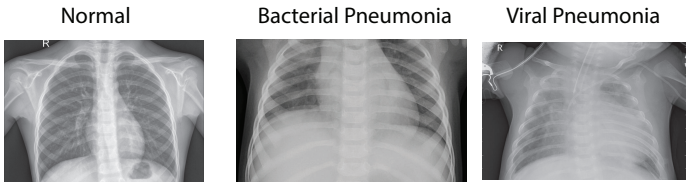


Figure 9: Examples of chest x-rays from patients.

Figure 9 shows some examples of images present in the database. A normal chest x-ray (left) shows clean lungs with no abnormal areas of opacification in the image. Bacterial pneumonia (center) usually shows focal lobar consolidation, for instance, in the right upper lobe (white arrows), while viral pneumonia (right) manifests with a more diffuse "interstitial" pattern in both lungs [3].

However, only two classes (normal and pneumonia) are considered in this paper. The division was as follows: 4,273 x-ray images from patients with pneumonia and 1,583 images from patients in normal conditions.

5.2 Data Augmentation

Data Augmentation is a manner to reduce overfitting in deep neural networks. This technique creates many synthetic examples of the same image, allowing better learning, which is offered with the artificial increase of information [33]. During the Data Augmentation process, each image goes through a series of transformations to generate new images, such as rotation, translation, and scale. It makes modifications to the images at each iteration during the training to make the networks not know the training data completely, so that the networks do not lose their ability to generalize, becoming specialist only in the training data [24]. Table 2 describes the data augmentation parameters that were used.

Table 2: Data augmentation parameters.

Transformation	Value
epsilon for ZCA whitening	1e-06
width shift range	0.1
height shift range	0.1
fill mode	nearest
horizontal flip	True

5.3 Testing Parameters

The database, with 5,856 images, was divided into training data (4,684 images) and test data (1,172 images). Some of them have different resolutions, which can be up to 1917x1432 pixels. It was necessary to standardize them so that they could

pass through the network, thus a standard resolution of 75x75 pixels was assumed. Some parameters used in the tests for all architectures were:

- Output layer activation function: Sigmoid;
- Batch Size (*batch*): 16;
- Activation function in hidden layers: ReLU (Rectified Linear Units);
- Optimizer: Adam;
- Number of epoch: 200;
- Dropout: 0.2;
- Learning rate: The initial learning rate is: 1x10^{-3}, multiplied by lower rates throughout the epoch, starting from epoch 81: 1x 10^{-1}/epoch 121: 1x 10^{-2}/epoch 161: 1x 10^{-3}/epoch 181: 0.5x 10^{-3};
- Cost Function: binary cross-entropy.

5.4 Evaluation Metrics

Some evaluation metrics were applied to improve the interpretation of the results, as synthesized in the following.

5.4.1 Confusion Matrix

A confusion matrix compares the model predictions with the true patterns. The main diagonal of the matrix represents classes that were correctly predicted (True Positive - TP and True Negative - TN), while elements of the secondary diagonal represent examples that were wrongly classified (False Positive - FP and False Negative - FN) [24, 34]. From the confusion matrix, it is possible to calculate the metrics to assess whether or not the algorithm achieved feasible results. Table 3 shows an example of confusion matrix in the context of the problem addressed in this paper.

Table 3: Example of a confusion matrix.

	Normal	**Pneumonia**
Normal	10 (TP)	5 (FP)
Pneumonia	4 (FN)	6 (TN)

5.4.2 Precision

Precision is intuitively the classifier's ability to not label a negative sample as positive [34], where TP is the number of true positives and FP is the number of false positives.

$$\text{precision} = \frac{TP}{TP + FP}. \tag{1}$$

5.4.3 Recall

Recall is intuitively the classifier's ability to find all positive samples [34], where FN is the number of false negatives.

$$\text{recall} = \frac{TP}{TP + FN}. \tag{2}$$

5.4.4 F1-score

The f1-score can be interpreted as a weighted average of precision and recall, in which a f1-score assumes its best value as 1 and the worst score as 0. The relative contribution of precision and recall to the f1-score is equal [34].

$$\text{f1-score} = 2 * \frac{precision * recall}{precision + recall}. \tag{3}$$

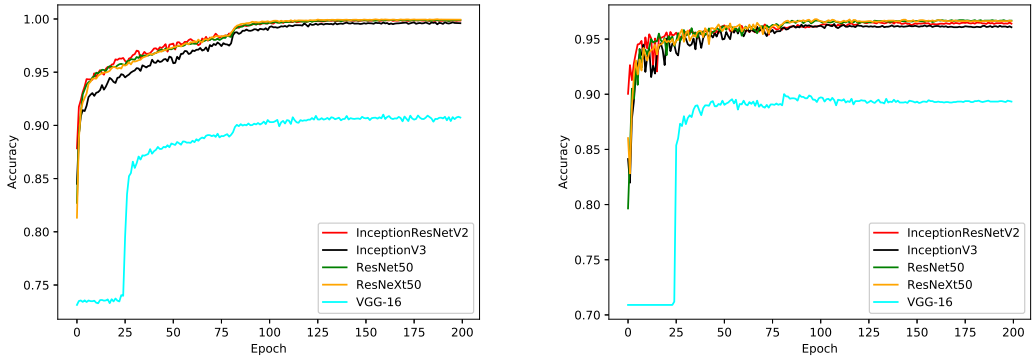
6. RESULTS AND DISCUSSION

Table 4 shows the precision, recall, and f1-score for each architecture, considering the average of ten simulations for each CNN. Based on these results, one can see that the InceptionResNetV2 obtained the best recall for the Normal class and precision for the Pneumonia class (93.95% and 97.52%, respectively). Thus, this CNN has a greater capacity to find positive examples and high accuracy when classifying x-ray images of patients with pneumonia. The ResNeXt50 was the architecture that obtained the best result in terms of precision and f1-score for the Normal class (94.62% and 94.25%, respectively) and in terms of recall and f1-score for the Pneumonia class (97.80% and 97.65%, respectively), that is, it has greater ability to find negative samples in addition to being more accurate when classifying x-ray images of patients in normal conditions. For both classes, ResNeXt50 obtained the best f1-score results, i.e., it is the most balanced architecture to acquire the best possible accuracy.

Table 4: Average and standard deviation for precision, recall, and f1-score from ten simulations.

Architecture	Class	precision	recall	f1-score
ResNet50	Normal	94.43% ±0.0102	93.87% ±0.0068	94.14% ±0.0062
	Pneumonia	97.49% ±0.0027	97.72% ±0.0044	97.60% ±0.0026
InceptionV3	Normal	93.20% ±0.0048	93.34% ±0.0114	93.27% ±0.0063
	Pneumonia	97.26% ±0.0045	97.20% ±0.0021	97.23% ±0.0024
VGG-16	Normal	82.80% ±0.0154	80.11% ±0.0103	81.42% ±0.0062
	Pneumonia	91.94% ±0.0034	93.15% ±0.0078	92.54% ±0.0033
InceptionResNetV2	Normal	93.86% ±0.0097	**93.95%** ±0.0075	93.90% ±0.0056
	Pneumonia	**97.52%** ±0.0030	97.47% ±0.0042	97.49% ±0.0023
ResNeXt50	Normal	**94.62%** ±0.0083	93.90% ±0.0084	**94.25%** ±0.0056
	Pneumonia	97.50% ±0.0033	**97.80%** ±0.0036	**97.65%** ±0.0022

Figure 10 demonstrates the accuracy of each architecture during the training (10a) and test (10b) stages. In general, it is possible to perceive a certain balance between the architectures, except for VGG-16, which is below the others in terms of performance. ResNeXt50, InceptionResNetV2, InceptionV3, and ResNet50 converged quickly, obtaining an accuracy greater than 90% in a few epochs (before epoch 25) and VGG-16 reached this percentage only after the epoch 75, because it is an architecture that has many trainable parameters that need to be adjusted.



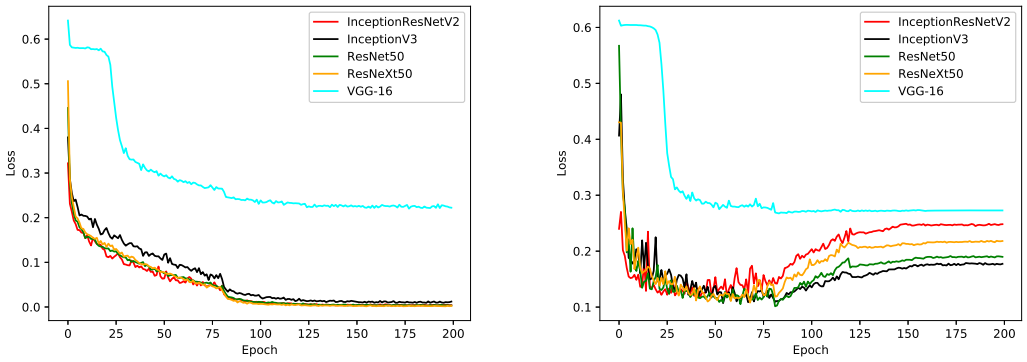
(a) Train. (b) Test.

Figure 10: Accuracy in the training and test from the average of ten simulations for each architecture.

Regarding the loss, as shown in Figure 11, it is interesting to notice that in both the training (11a) and test (11b) stages, the loss functions start with values around 0.6, which are gradually minimized. Except for VGG-16, the architectures have similar behaviors in training, but there is some divergence during the testing phase. The high peaks in the average behavior of architectures are due to the learning rate that is optimized and decreased over time. Besides, one can infer that the loss function for VGG-16 has a distinct behavior, as the curve has premature convergence, which corroborates the relative poor performance in terms of classification against the other CNNs.

Figure 12 shows the confusion matrix for each architecture, where the percentages of TP were: InceptionResNetV2, 92.66%; InceptionV3, 95.01%; ResNet50, 94.13%; ResNeXt50, 92.96%; and VGG-16, 80.05%. The percentages of TN were: InceptionResNetV2, 97.71%; InceptionV3, 97.11%; ResNet50, 98.31%; ResNeXt50, 98.07%; and VGG-16, 92.17%. Thus, one can see that the InceptionV3 and ResNet50 models are recommended for classifying images from the Normal and Pneumonia classes, respectively.

Regarding the errors, it is possible to observe the percentages of FP as: InceptionResNetV2, 7.33%; InceptionV3, 4.98%; ResNet50, 5.86%; ResNeXt50, 7.03%; and VGG-16, 19.94%. The percentages of FN were: InceptionResNetV2, 2.28%; InceptionV3, 2.88%; ResNet50, 1.68%; ResNeXt50, 1.92%; and VGG-16, 7.82%. Thereby, the InceptionV3 model minimized the False Positive errors, while the ResNet50 model reduced the False Negative errors.



(a) Train. (b) Test.

Figure 11: Loss from the average of ten simulations in the training and test for each architecture.

7. FINAL CONSIDERATIONS AND FUTURE WORK

The five architectures used in this research obtained good results, with accuracy above 90%, with InceptionResNetV2 and ResNeXt50 achieving superior results in terms of precision, recall, and f1-score. ResNeXt50 and ResNet50 were able to reduce False Negative errors and maximize True Positive and True Negative hits.

From this research, one can observe that the architectures that apply optimization mechanisms which prevent the vanishing gradients tend to present better results, such as ResNet50. The combination of any other architectures with ResNet obtained results superior to the original architecture, due to the fact that ResNet uses the residual blocks that prevent the vanishing gradients. The VGG-16 showed results below the average of the others because it is an older architecture and has many parameters to be trained.

From this research, new possibilities can be reached in relation to the classification of x-ray images; new images can be collected, making it possible to increase the classes from two (normal and pneumonia) to four (normal, viral pneumonia, bacterial pneumonia and fungal pneumonia), yielding more robust models for image classification.

A future work that can be considered is the use of other architectures, such as MobileNet, VGG-19, Xception, and DenseNet, to develop a new comparative study. In this manner, it is possible to improve the learning process and reduce False Positive and False Negative errors, generating an ideal model for classifying x-ray images. Techniques, such as transfer learning [35], can be used to assist in the classification of x-ray images with pneumonia using pre-trained weights, as well as for other types of diseases, such as bone cancer, breast cancer, tumors, and so on.

Another possibility is the use of the National Institutes of Health Chest X-Ray database to improve studies with these and other CNN architectures, as mentioned before. It is a broad database, both of images (112,120 images) and classes (15 classes), using the multi-class and multi-label classification paradigm.

REFERENCES

[1] S. S. Haykin *et al.*. *Neural networks and learning machines/Simon Haykin.*. New York: Prentice Hall,, 2009.

[2] T. Karnkawinpong and Y. Limpiyakorn. “Chest X-Ray Analysis of Tuberculosis by Convolutional Neural Networks with Affine Transforms”. In *Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence*, pp. 90–93. ACM, 2018.

[3] D. S. Kermany, M. Goldbaum, W. Cai, C. C. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan *et al.*. “Identifying medical diagnoses and treatable diseases by image-based deep learning”. *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018.

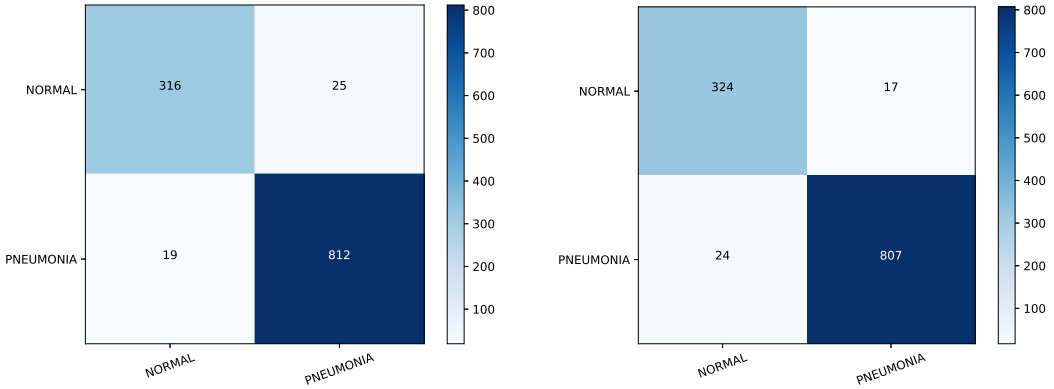
[4] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri and R. M. Summers. “Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases”. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2097–2106, 2017.

[5] W. H. Organization. “Pneumonia”, aug 2019.

[6] C. Neto and S. A. de La Hidalga. “Reconhecimento de tumores cerebrais utilizando redes neurais convolucionais”. 2017.

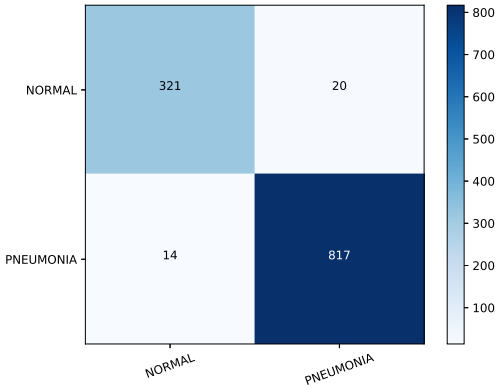
[7] M. Haloi, K. R. Rajalakshmi and P. Walia. “Towards Radiologist-Level Accurate Deep Learning System for Pulmonary Screening”. *arXiv preprint arXiv:1807.03120*, 2018.

[8] G. Liang and L. Zheng. “A transfer learning method with deep residual network for pediatric pneumonia diagnosis”. *Computer Methods and Programs in Biomedicine*, 2019.

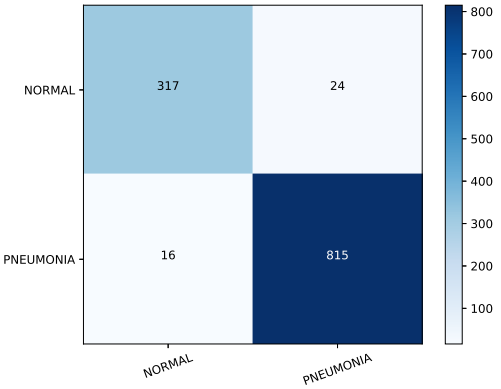


(a) InceptionResNetV2.

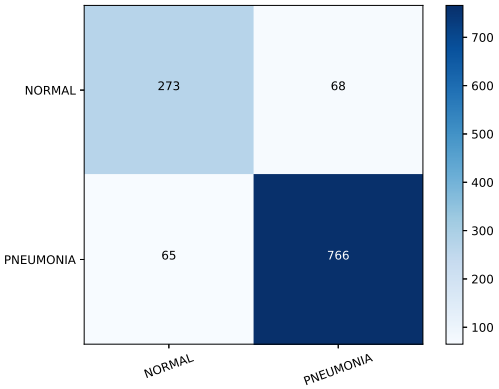
(b) InceptionV3.



(c) ResNet50.



(d) ResNeXt50.



(e) VGG-16.

Figure 12: Confusion Matrices.

[9] O. Stephen, M. Sain, U. J. Maduh and D.-U. Jeong. "An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare". *Journal of healthcare engineering*, vol. 2019, 2019.

[10] T. Liu, C. Rosenberg and H. A. Rowley. "Clustering billions of images with large scale nearest neighbor search". In *2007 IEEE Workshop on Applications of Computer Vision (WACV'07)*, pp. 28–28. IEEE, 2007.

[11] G. D. Juraszek *et al.*. "Reconhecimento de produtos por imagem utilizando palavras visuais e redes neurais convolucionais". 2014.

[12] I. Arel, D. C. Rose, T. P. Karnowski *et al.*. "Deep machine learning-a new frontier in artificial intelligence research". *IEEE computational intelligence magazine*, vol. 5, no. 4, pp. 13–18, 2010.

[13] W. Mousser and S. Ouadfel. "Deep Feature Extraction for Pap-Smear Image Classification: A Comparative Study". In *Proceedings of the 2019 5th International Conference on Computer and Technology Applications*, pp. 6–10. ACM, 2019.

[14] A. Santos, K. Aires, R. Veras, V. Uchôa and L. Santos. "Uma Abordagem de Classificação de Imagens Dermatoscópicas Utilizando Aprendizado Profundo com Redes Neurais Convolucionais". In *Anais do XVII Workshop de Informática Médica*. SBC, 2017.

[15] O. M. Parkhi, A. Vedaldi, A. Zisserman *et al.*. "Deep face recognition." In *bmvc*, volume 1, p. 6, 2015.

[16] Z. Shi and L. He. "Current status and future potential of neural networks used for medical image processing". *Journal of multimedia*, vol. 6, no. 3, pp. 244, 2011.

[17] J. Kawahara, A. BenTaieb and G. Hamarneh. "Deep features to classify skin lesions". In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pp. 1397–1400. IEEE, 2016.

[18] K. Simonyan and A. Zisserman. "Very deep convolutional networks for large-scale image recognition". *arXiv preprint arXiv:1409.1556*, 2014.

[19] A. Vedaldi and K. Lenc. "Matconvnet: Convolutional neural networks for matlab". In *Proceedings of the 23rd ACM international conference on Multimedia*, pp. 689–692. ACM, 2015.

[20] P. S. Dainez and E. C. L. Dainez. "Rede Neural Artificial Aplicada em um Sistema de Auxílio ao Rastreio de Depressão e de Qualidade de Vida de Idosos". *Learning & Nonlinear Models*, vol. 13, no. 2, pp. 67–72, 2015.

[21] K. He, X. Zhang, S. Ren and J. Sun. "Deep residual learning for image recognition". In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[22] M. Rizwan. "Residual Networks (Resnets)", oct 2018.

[23] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna. "Rethinking the inception architecture for computer vision". In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

[24] R. E. V. da Silva. "Um estudo comparativo entre redes neurais convolucionais para a classificação de imagens", 2018.

[25] A. Nivrito, M. Wahed, R. Bin *et al.*. "Comparative analysis between Inception-v3 and other learning systems using facial expressions detection". Ph.D. thesis, Brac University, 2016.

[26] C. Szegedy, S. Ioffe and V. Vanhoucke. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning". *CoRR*, vol. abs/1602.07261, 2016.

[27] S. Xie, R. Girshick, P. Dollár, Z. Tu and K. He. "Aggregated residual transformations for deep neural networks". In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

[28] N. Ketkar. "Introduction to keras". In *Deep Learning with Python*, pp. 97–111. Springer, 2017.

[29] F. Chollet *et al.*. "Keras", 2015.

[30] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*. "Tensorflow: A system for large-scale machine learning". In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, 2016.

[31] T. Carneiro, R. V. M. Da Nóbrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque and P. P. Reboucas Filho. "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications". *IEEE Access*, vol. 6, pp. 61677–61685, 2018.

[32] M. Gutoski, L. T. Hattori, N. M. R. Aquino, H. C. M. Senefonte, M. Ribeiro, A. E. Lazzaretti and H. S. Lopes. "Quantitative Analysis of Deep Learning Frameworks". *Learning & Nonlinear Models*, vol. 15, no. 1, pp. 45–52, 2017.

[33] P. Y. Simard, D. Steinkraus, J. C. Platt *et al.*. "Best practices for convolutional neural networks applied to visual document analysis." In *Icdar*, volume 3, 2003.

[34] G. A. d. S. Soares, G. J. F. d. Silva, H. T. Macedo, B. O. P. Prado and L. N. Matos. "Reconhecimento de estados dos olhos utilizando máquinas de aprendizado profundo a partir de ondas cerebrais". *Revista de Sistemas e Computação-RSC*, vol. 9, no. 1, 2019.

[35] L. F. Brunialti, S. M. Peres, C. A. M. Lima and C. Boscarioli. "Aprendizado por Transferência para Aplicações Orientadas a Usuário". *Learning & Nonlinear Models*, vol. 11, no. 2, pp. 56–73, 2013.