IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 28, 2022, accepted February 20, 2022, date of publication February 24, 2022, date of current version March 4, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3154061

Data Augmentation and Transfer Learning for Brain Tumor Detection in Magnetic Resonance Imaging

ANDRÉS ANAYA-ISAZA[1,2] **AND LEONEL MERA-JIMÉNEZ**[2,3]
[1]Faculty of Engineering, Pontificia Universidad Javeriana, Bogota 410001, Colombia
[2]INDIGO Research, Bogota 410010, Colombia
[3]Faculty of Engineering, Universidad de Antioquia, Medellin 050010, Colombia

Corresponding author: Leonel Mera-Jiménez (leonel.mera@udea.edu.co)

This work was supported by INDIGO Technologies Division Research (https://indigo.tech/).

ABSTRACT The exponential growth of deep learning networks has allowed us to tackle complex tasks, even in fields as complicated as medicine. However, using these models requires a large corpus of data for the networks to be highly generalizable and with high performance. In this sense, data augmentation methods are widely used strategies to train networks with small data sets, being vital in medicine due to the limited access to data. A clear example of this is magnetic resonance imaging in pathology scans associated with cancer. In this vein, we compare the effect of several conventional data augmentation schemes on the ResNet50 network for brain tumor detection. In addition, we included our strategy based on principal component analysis. The training was performed with the network trained from zeros and transfer-learning, obtained from the ImageNet dataset. The investigation allowed us to achieve an F1 detection score of 92.34%. The score was achieved with the ResNet50 network through the proposed method and implementing the learning transfer. In addition, it was also concluded that the proposed method is different from the other conventional methods with a significance level of 0.05 through the Kruskal Wallis test statistic.

INDEX TERMS Artificial intelligence, biomedical imaging, cancer, machine learning, medical diagnostic imaging.

I. INTRODUCTION

Since the end of the 20th century and the beginning of the 21st century, we have witnessed the new industrial revolution, the second informatics revolution [1]. The emerging developments and technological advances have allowed us to create increasingly powerful tools with incredible performances in different areas, where medicine could not be the exception [2]. Advances range from simple tasks to tasks so complex that they were usually performed by professionals or experts [2], [3]. These advances are largely thanks to artificial intelligence (AI), one of the most awaited paradigms since several decades ago [4]. It is perhaps a little challenging to define artificial intelligence since many authors establish intelligence as the ability to generate a response to a stimulus or achieve a goal in a specific environment [5]. In this sense, artificial intelligence can range from the most straightforward systems to highly complex processes that resemble the cognitive processes performed by the human brain [6], [7]. The latter is the desired approach, where deep learning (DL) has managed to address some of these processes, even surpassing human performance in some tasks [8]–[10].

Moreover, DL is one of the fastest-growing topics in recent years, arousing interest in various research areas that rely on manual, extensive, or tedious processes, such as medicine [11], [12]. Besides, DL artificial neural networks have advantages that make them even more attractive. For example, DL networks do not require prior feature extraction and can be used directly on the raw data [13]. Moreover, despite the complexity of the tasks, the network model is usually governed by a few mathematical expressions [14]. While the model becomes complex because of the number of layers that constitute it, the forte of deep learning is focused on task performance and not on statistical or mathematical inference, i.e., for most researchers, DL models can be

The associate editor coordinating the review of this manuscript and approving it for publication was Genoveffa Tortora.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

treated as black boxes that need input data and labels to replicate tasks (Supervised Learning) [15]. These examples are detection, classification, segmentation, and prediction, among the most common applications in medicine [16], [17]. -Unfortunately, not everything is so simple in DL-. One of the main challenges is having a large amount of data to efficiently train AI models in different tasks. This constraint is the most critical point, especially in medicine, where access to images is limited by cost or few study subjects. Besides, the images require the authorization of the subjects, who may refuse even if their data are anonymized [18]. Consequently, strategies to synthesize or augment data are becoming increasingly common in studies with small data sets [19].

On the other hand, artificial intelligence (AI) has taken a vital role in radiology, allowing it to automatically address tasks such as lesion detection or image quantification [18]. Moreover, due to high effectiveness and reduced processing times, AI (especially deep learning) has been highly involved in cancer pathology or brain tumors, one of the most recurrent diseases worldwide [20]. Brain tumors are a type of cancer manifested by excessive and uncontrolled growth of abnormally functioning cells [21]. The damage to the cells is generated by different factors that can range from genetic (increasing the risk of suffering from this pathology) to external factors such as chemical substances or exposure to high radiation sources [22]. In general, tumors are classified into heterogeneous neoplasms that include differentiable lesions (e.g., meningiomas) or highly invasive and poorly differentiable lesions such as multiform gliomas [23]. Glioma has the highest mortality rate among brain tumors, manifesting with increased progression of pathology. Statistics show that glioma accounts for almost 80% of malignant tumors [24], generating a 5-year survival rate of less than 21% in people older than 40 years [25]. However, early detection leads to a significant reduction in these statistics [26]. Fortunately, great efforts are invested today to address this, and other targets related to brain cancer [27]. Research is conducted using different tools, including DL, a science that has been very popular in recent years in the radiological field. For example, so far in 2021, DL research related to brain cancer can be found, such as Radiation therapy planning of head and neck cancer patients [28], automatic diagnosis of brain tumors [29], detection and classification of brain tumors [30]–[33], diagnostic feasibility assessment with DL networks [34], detection of brain metastases [35], prediction of survival in patients with infiltrating gliomas [36], the prognosis of glioblastoma multiforme [37], analysis for diagnostic biomarkers of glioma [38], segmentation of brain tumors [39]–[42], segmentation in dosimetry in organs at risk [43], and denoising to improve quality in subjective imaging [44].

Recent research shows promising findings and results, covering many applications in favor of brain tumor detection and treatment. However, despite the good results highlighted by the authors, few investigations have validity in the real clinical context due to serious limitations. Mainly, the authors highlight the limited access or the small amount of data for training the models, preventing the generalization of the results. For example, Olin *et al.* state that the models used were trained with small data sets for head and neck patients, limited to a study of no more than 800 scans [28]. Similarly, Jayachandran *et al.* have only 775 patients with glioblastomas [34]. Similarly, Amemiya *et al.* work with 127 patients, stating that the data are small, which would imply a better performance if the number of data is augmented [35]. For their part, Tandel *et al.* are limited to 130 patients with brain tumors; however, they avoid this drawback by using transfer learning and augmenting the data with image scaling and rotation [30]. Similarly, Jiang *et al.* increase the number of images through flipping, scaling, and smoothing [39]. Similarly, Wang *et al.* use rotation, flipping, image warping, and color (contrast) change through gamma function [41]. In general, most authors performed the applications with a small data set; however, they did not implement any data augmentation strategy or learning transfer. Examples of these are: Menze, Al-Saffar, Khairandish, Islam, Song, Poel, Yan, and Wong *et al.* [29], [31], [32], [36]–[38], [43], [44].

The presented literature clearly shows the need to augment the number of training data due to the limited available data set. Moreover, the few studies that use data augmentation do so without reporting which strategy is more efficient. Therefore, in this work, we explore the different data augmentation strategies on the performance of the ResNet50 network in brain tumor detection in magnetic resonance images.

- This research work offers the following novel contributions:
- A review of conventional data augmentation methods is presented.
- A new data augmentation method based on principal component analysis is proposed.
- A comparative framework between different data augmentation methods is proposed.
- The effect of transfer learning on the performance of the convolutional neural network ResNet50 is compared.
- The results are evaluated through the non-parametric Kruskal Wallis test, based on the distribution of means of the data.

A comparison between the activation maps of the ResNet50 layers under the similarity coefficient is presented with centered kernel alignment.

II. MATERIALS AND METHODS

A. DATASET

The investigation was based on The Cancer Genome Atlas Low-Grade Glioma (TCGA-LGG) database [45], [46]. The set has 110 participants and three types of image acquisition sequences, with fluid-attenuated inversion-attenuated inversion recovery (FLAIR) imaging being the sequence of choice for data augmentation. The images are axial slices of size

256 × 256 in uint8 format, i.e., images with 8-bit unsigned integer data.

B. DATA PREPROCESSING

The images were only reformatted and normalized, leaving the intensity values on the 0 to 1 scale in float32 format. Deep learning methods are generally designed to work on the raw data [47]–[49]; therefore, no further preprocessing was performed on the images.

C. CONVOLUTIONAL NEURAL NETWORK

There are many deep learning neural networks, and this approach is of great interest since greater depth allows the network to perform more complex tasks [50]. However, the increase in depth poses two main problems. First, deeper networks require a larger number of training parameters, hence a larger dataset to arrive at a high-performance network, and second, depth limits training due to gradient fading [51]. In this research, the different data augmentation strategies are used to solve the first drawback and, to solve the second one, the ResNet50 network was chosen [52]. The network is described in detail in appendix A.

D. DATA AUGMENTATION

The inherent need for large amounts of data in deep learning networks has encouraged the development of many strategies ranging from simple transformations such as geometric transformations to complex images composed of mosaics. Among the most commonly used techniques [19], [53], are the following basic techniques:

- Translation [54], [55].
- Rotation [55], [56].
- Flip
- Resizing
- Distortion [57]–[60].
- Cropping [61].
- Image overlay [19].
- Noise injection [62].
- Color space [63].
- Linear filters [64], [65].
- Random deletion of frames [66].

The methods are classified as basic and/or deformable and represent about 86% of the data augmentation methods applied in medical imaging for deep learning [53]. Each method listed is described in detail in Appendix B.

1) PCA-BASED AUGMENTATION (PROPOSED METHOD)

Principal component analysis (PCA) is generally used to reduce the dimensions of a data set or even eliminate noise if it is used as an encoder-decoder [67]–[69]. The method takes a series of samples or observations and creates new components generated as the linear combinations of the first ones. The components are generated hierarchically, and each component represents a percentage of the variability of the data, where the first component z_1 has the largest percentage, and each new component has a smaller percentage than the previous one. Mathematically, the first principal component has the form expressed in Equation (1) or (2).

$$z_1 = u_{11}x_1 + u_{12}x_2 + u_{13}x_3 + \ldots + u_{1m}x_m \tag{1}$$

$$z_1 = U_1 \cdot X \tag{2}$$

In other words, let X be an observation of m variables, i.e., $X \in R^m$. The observation can be represented from a smaller number of latent variables Z, as shown in Equation (3).

$$Z = WX \tag{3}$$

$$W = \begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_n \end{pmatrix} = \begin{pmatrix} u_{11} & u_{12} & \ldots & u_{1m} \\ u_{21} & u_{22} & \ldots & u_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \ldots & u_{nm} \end{pmatrix} \tag{4}$$

where, Z is the vector of n principal components, with $n < m$. In other words, Z^t equals $(z_1, z_2, z_3, \ldots, z_n)$, with each principal component z_i being the linear combination of the original m variables and W the matrix of the coefficients of these linear combinations, which are calculated following the following considerations: For the first component z_1 the maximum variance subject to the constraint of Equation (5) must be satisfied.

$$U_1 U_1^t = \sum_{i=1}^{m} u_{1i}u_{1i} = 1 \tag{5}$$

Subsequent components are calculated under the same reasoning, considering that the new components must be orthogonal to the previous ones, i.e., the i-th component must fulfill the restriction of Equation (6).

$$U_i U_j^t = 0 \quad \forall j < i \tag{6}$$

For the case of images, the reasoning is the same. However, each image would represent an observation X and each pixel of the image a different feature. Thus, for an image of 128 × 128 there would be 16.384 features. Therefore, it is possible to represent the pixels of an image in a smaller number of features while preserving the higher variability of the images.

The transformation by PCA to several latent variables is reversible, i.e., the original variables can be obtained from the principal components, and the greater the number of components taken, the greater the similarity in the reconstruction of the original variables. Generally, the reconstruction of images with a smaller number of components is used to eliminate noise because components associated with such noise are eliminated [67]. The process is based on finding the projections of the components on the original centered space, as shown in Equation (7).

$$X = W^t \hat{Z} = W^t W \hat{X} \tag{7}$$

where, $\hat{X}$ is the observation with the m variables centered with respect to their means μ_i $(i = 1, \ldots m)$, for the case of several observations.

As mentioned above, PCA can be used to eliminate noise by taking a smaller number of principal components for the

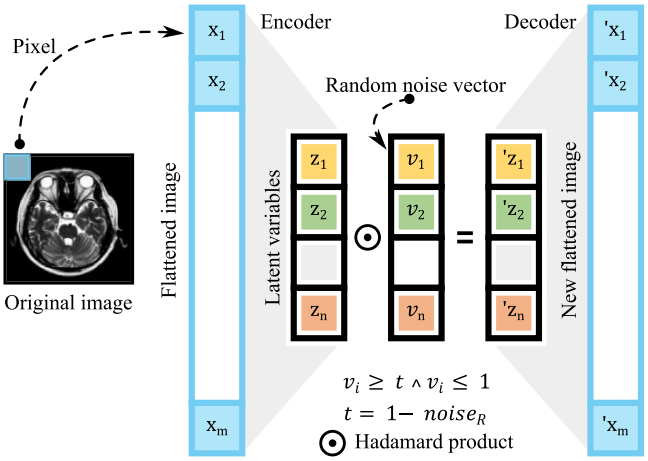

FIGURE 1. Graphical description of the proposed new method for data augmentation based on PCA.

reconstruction. In other words, eliminating the last principal components would preserve most of the explained variance and, therefore, the fundamental essence of the image would be preserved. In this order of ideas, altering the principal components with random noise would imply the partial modification of the image, preserving its primary attributes, i.e., it could be possible to generate new images from a reference image.

Based on the above considerations, it was proposed to generate images as follows: The original images were flattened to vectors of 16, 384 features. The features were projected into a latent space of lower dimensionality through PCA. Each vector was multiplied pointwise (Hadamard Product) by a random noise vector V_r with the same dimensions but with values from a threshold t to 1 ($V_r \in [t, 1]$). The threshold was determined as $t = 1 - noise_R$, where $noise_R$ is the proportion of noise added. For example, if $noise_R$ equals zero, the values of V_r would be constrained to 1, implying that the latent variables would not be altered. Finally, the modified latent variables Z' were used to generate the new features through the inverse transformation. The process is exemplified in Figure 1.

Mathematically, the model of the new images would be given by the expression of Equation (8).

$$X' = W^t Z' \tag{8}$$
$$Z' = (\hat{Z} \odot V_r) \tag{9}$$

E. TRANSFER LEARNING

As mentioned above, DL networks need a large amount of training data due to the high number of parameters. In this sense, transfer learning is another widely used method to initialize the model weights, avoiding training from zeros or random distributions. The process consists of taking a network and training it with an extensive database, allowing filters to take the weights to create the complex activation maps associated with that dataset. Generally, if the database is large enough, the network learns the task with a high degree of generalization. The attribute can be retained for an equivalent task with another dataset, and the network would generate good results if subsequently trained with the new data, even if the data is sparse [70], [71]. Following this order of ideas, data augmentation methods were trained from zeros and implementing transfer learning with the ResNet50 network and the ImageNet natural image database [72].

F. LOSS FUNCTION

Although many loss functions exist, cross-entropy remains one of the most reported and used for the case of two-class classifications [73], as is this case. Precisely, the function measures the difference between two probability distributions, calculating the entropy associated with each class or element. The concept can be applied to images, taking each pixel as one of two distribution elements (e.g., healthy tissue and tumor) [74]. The binary cross-entropy (L_{BCE}) is defined mathematically, as shown in Equation (10).

$$L_{BCE}\left(y, \hat{y}\right) = -\left(y \log\left(\hat{y}\right) + (1-y) \log\left(1-\hat{y}\right)\right) \tag{10}$$

where, y is the actual data set and $\hat{y}$ is the predicted set.

G. EVALUATION METRICS

As an important part of an objective comparison of the models used, our approach was based on four evaluation metrics: Accuracy, Sensitivity, Specificity, F_1 score, and Precision. The metrics are expressed as shown in Equations (11) through (15) [75]–[77].

$$Accuracy = \frac{TP + TN}{TN + TP + FP + FN} \tag{11}$$
$$Sensitivity = \frac{TP}{TP + FN} \tag{12}$$
$$Specificity = \frac{TN}{TN + FP} \tag{13}$$
$$F1_{score} = \frac{2TP}{2TP + FP + FN} \tag{14}$$
$$Precision = \frac{TP}{TP + FP} \tag{15}$$

The above metrics are expressed in terms of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). In addition, for this specific case, the metrics represent the following observations:

Accuracy: The ability of a network to correctly classify the different classes, i.e., tumor and non-tumor.

Sensitivity: The ability of a network to classify actual tumors.

Specificity: The ability of a network to correctly classify real non-tumor images.

F_1 Score: The ability to correctly identify the different classes in proportion to the number of classes.

H. STATISTICAL ANALYSIS

The Kruskal Wallis test was used for statistical estimation between groups, which evaluates whether two or more samples belong to the same distribution based on the median

of these samples. The test uses the null hypothesis with the assumption that all samples come from the same distribution. Then, for a p value less than 0.05, it would imply that the null hypothesis is false and, therefore, a statistically significant difference would be established between the two groups tested. Note that the value of 0.05 or significance level can have a lower or higher value. However, this value is the most accepted since it represents only 5% of concluding that there is a difference when there is none [78]. The method, assuming k groups with n observations, defines the H statistic given by the mathematical expression of Equation (16).

$$H = \frac{12}{N(N+1)} \sum_{i=1}^{k} n_i (\overline{r_{i.}} - \overline{r})^2 \tag{16}$$

$$\overline{r_{ij}} = \frac{\sum_{i=1}^{n_i} r_{ij}}{n_i} \tag{17}$$

where, n_i is the number of observations in the i-th group, N is the total number of observations in the two groups, r_{ij} is the rank of the i-th observation over the j-th observation among all observations and k is the number of groups [79], [80].

I. CENTERED KERNEL ALIGNMENT (CKA)

Finally, to understand the behavior of neural networks as a function of different layers, the similarity between layers was included through the centered kernel alignment method [81]. Particularly, CKA takes two feature maps as inputs and calculates the normalized similarity, as shown in Equation (18).

$$CKA(K, L) = \frac{HSIC(K, L)}{\sqrt{HSIC(K, K) HSIC(L, L)}} \tag{18}$$

where, K and L are the similarity matrices of any two feature maps (see Equation (A.1)), HSIC is the Hilbert-Schmidt independence criterion for similarity based on the dot product [81], [82].

J. EXPERIMENTAL DESIGN

The different data augmentation strategies were compared by training the ResNet50 network with the TCGA-LGG database. The data were normalized and split into training and validation data. The ResNet50 network was trained from scratch and implemented transfer learning from the trained network with the ImageNet database. Each training was executed with the k-folds method using 10 folds. The network was used under the binary cross-entropy loss function. In addition, the performance of the network during training was validated with the accuracy metric. Subsequently, the network was evaluated with the F_1 score, accuracy, sensitivity, specificity, and precision metrics through the test data.

It is worth noting that the network was run an average of 40 times under the following hyperparameters:

- Loss function: binary cross-entropy.
- Number of epochs: 50
- Optimizer: Adadelta
- Batch size: 10
- Initialization of weights: Uniform Glorot
- Bias initialization: Zeros

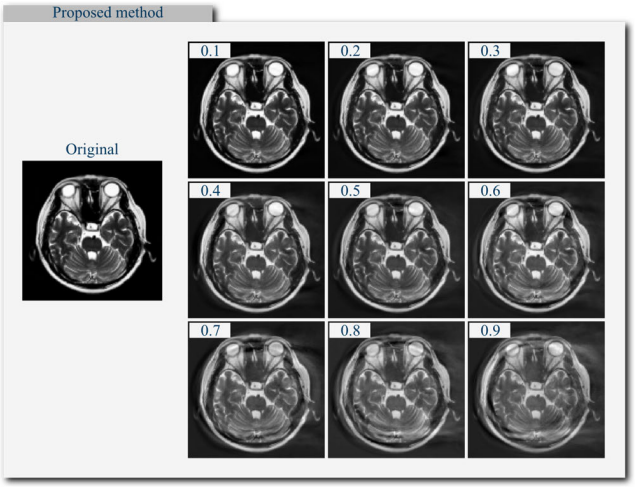

FIGURE 2. Image generation through PCA-based data augmentation.

Finally, the different configurations were compared through the Kruskal Wallis statistical model, where the p-value between these configurations was calculated to establish statistically significant differences. In addition, the similarity matrices generated by the centered kernel alignment method were also compared. The ResNet50 architecture was modeled with the main Keras and TensorFlow libraries under the Python programming language. The execution was performed on the Colab platform configured with 25 GB of Ram and Tesla T4 GPU.

The implemented codes are publicly available in the following GitHub repository: (https://github.com/Qsinap/Data_augmentation_with_PCA).

III. RESULTS

Initially, the images were generated through the proposed method based on principal component analysis (PCA). The reconstruction of the images with different noise ratios is illustrated in Figure 2. The results clearly show that the images retained part of their spatial characteristics, even for the case with a high percentage of noise (0.9 Noise equivalent to 90%). Consequently, we used the images with 90% random noise for data augmentation in this work.

As mentioned above, the network was trained under the different data augmentation methods, with and without transferring the weight values (Transfer Learning). Therefore, the results shown below obey both cases. It should be clarified that the tables and spider graphs are shown in percentage values, while the box-and-whisker and training figures are given in their fractional equivalents, i.e., with values from 0 to 1. Table 1 shows the maximum values achieved by the data augmentation methods. Additionally, this is ordered from highest to lowest, taking the F1 score as a reference. The results show that the proposed PCA-based method achieved the maximum values in both cases, i.e.,

TABLE 1. Maximum results of the 12 data augmentation methods with and without transfer learning.

Maximum values - Training from zero						Maximum values - Training with transfer learning					
	F1	Acc	Sen	Spe	Pre		F1	Acc	Sen	Spe	Pre
Based on PCA	87,56	86,50	98,00	98,00	96,92	Based on PCA	92,38	92,00	100,00	100,00	100,00
Cropping	83,96	83,00	89,00	89,00	87,50	Flip	90,38	90,00	98,00	98,00	97,44
Color space	82,79	81,50	92,00	92,00	88,89	Distortion	90,23	89,50	98,00	98,00	97,47
Translation	82,19	80,50	92,00	92,00	88,57	Cropping	90,00	89,00	99,00	99,00	98,75
Flip	81,55	81,00	88,00	88,00	85,19	Color space	89,76	89,50	99,00	99,00	98,65
Distortion	81,48	80,00	90,00	90,00	86,11	Translation	89,55	89,50	97,00	97,00	95,89
Resizing	81,20	79,00	95,00	95,00	92,42	Random frame deletion	89,32	89,00	98,00	98,00	97,47
Linear filters	80,56	79,00	92,00	92,00	88,73	Resizing	89,11	89,00	96,00	96,00	94,87
Rotation	80,37	80,00	88,00	88,00	85,19	Linear filters	88,48	87,50	98,00	98,00	96,97
Random frame deletion	80,19	79,50	88,00	88,00	85,00	Rotation	87,67	86,50	97,00	97,00	95,83
Overlapping	79,11	77,00	89,00	89,00	85,33	Overlapping	85,85	85,00	94,00	94,00	91,30
Noise addition	76,52	75,00	96,00	96,00	90,24	Noise addition	85,20	83,50	97,00	97,00	94,92

F1: F1_score, Acc: Accuracy, Sen: Sensitivity, Spe: Specificity and Pre: Precision.

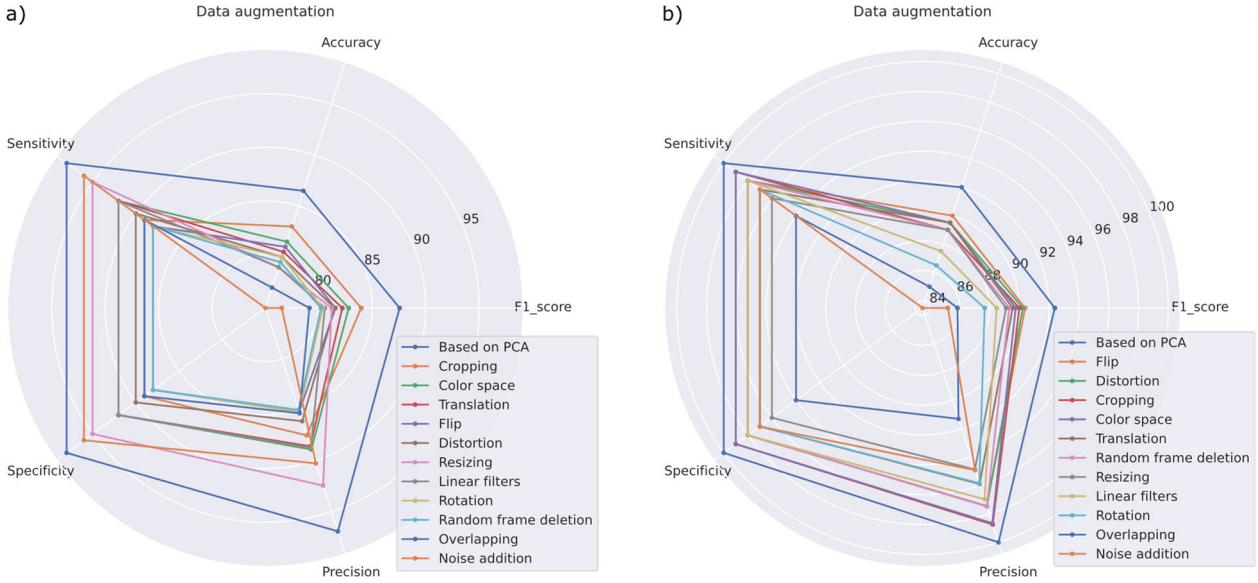

FIGURE 3. Spider graph with the five network evaluation metrics. Values obtained with the test data for network a) without learning transfer and b) with learning transfer.

with the model weights initialized at zero and learning transfer. Additionally, it can be seen that the scores reached higher values with learning transfer in all data augmentation methods, showing the effectiveness of such a strategy.

Similarly, Figure 3 shows the results of Table 1, being possible to observe that some methods presented similar behaviors. For example, random frame removal, overlapping, and noise addition had relative values in all five metrics. On the other hand, the proposed method is highly effective in both cases, i.e., without Transfer Learning and with Transfer Learning, generating the largest pentagons.

Additionally, it is worth noting that the trend in the scoring order of data augmentation methods was partially preserved, i.e., the proposed method generated the best results in the two cases. Similarly, noise addition and superposition maintained their positions, being the worst-performing strategies. In fact, the results show a maximum variation of up to 3 positions, where Flip, Distortion, and Random frame deletion, moved up three positions for the case with transfer learning.

Figure 4 presents the distributions of the 40 runs for each data augmentation method. Results were generated with the test data for the F1 score, accuracy, sensitivity, and specificity metrics. The distributions presented scores above 0.5, and it is even observed that the limits of the distributions reached values close to 1, demonstrating the effectiveness of the network combined with the data augmentation strategies. Additionally, the figure shows that the proposed method presented a compact distribution with the interquartile ranges with a more significant upward trend than the other methods. The behavior of the proposed method was maintained in

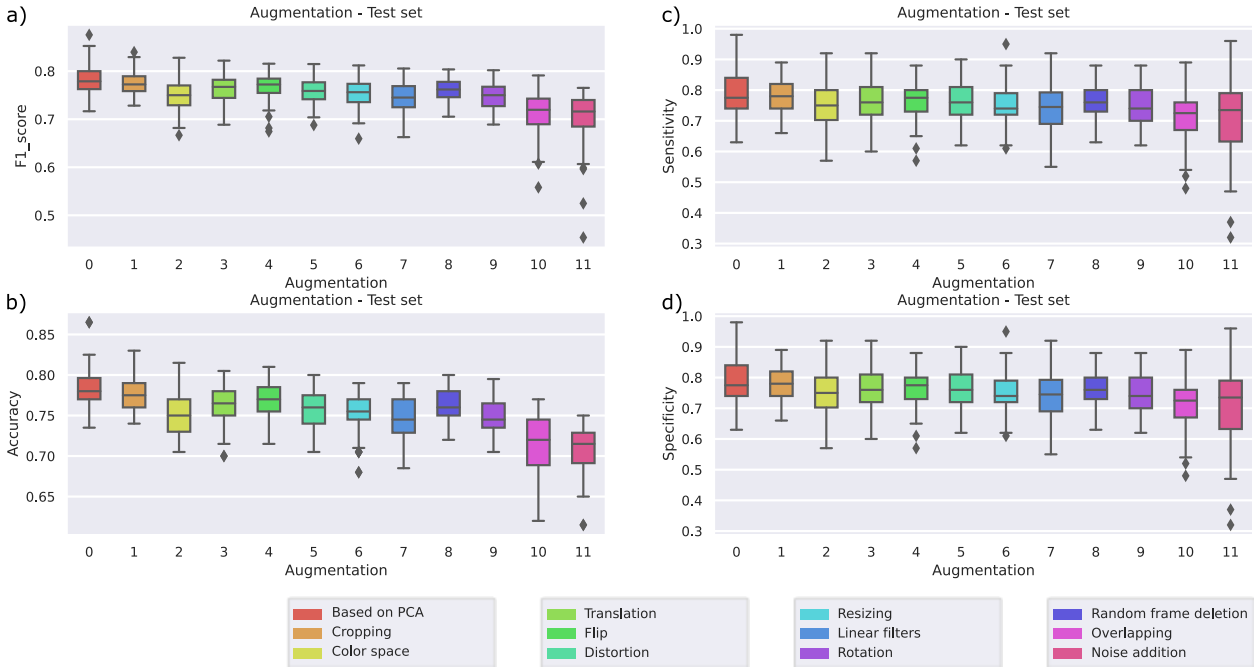

FIGURE 4. Score distributions of the metrics of a) F1 score, b) accuracy, c) sensitivity and d) specificity as a function of the 12 data augmentation methods. Network without learning transfer.

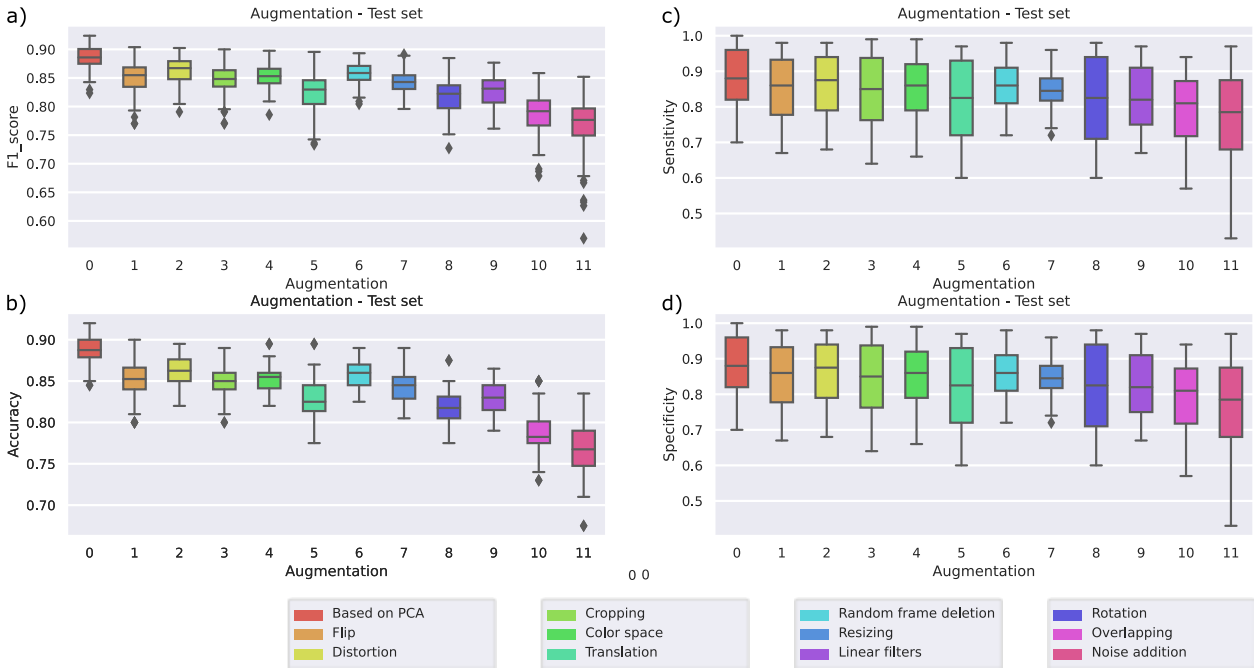

FIGURE 5. Score distributions of the metrics of a) F1 score, b) accuracy, c) sensitivity and d) specificity as a function of the 12 data augmentation methods. Network with learning transfer.

all four metrics, where it reached values close to 1 in the sensitivity and specificity metrics.

Similarly, Figure 5 presents the distributions of the 40 runs for each data augmentation method, but with learning transfer. In particular, the distributions generated with the learning transfer presented an upward shift, i.e., the results improved in all four metrics. Additionally, the figure shows that the proposed method presents the best distribution. Therefore, the proposed method is more likely to obtain a network with better performance.

The results presented better scores with the proposed method; therefore, only the training and similarity matrices

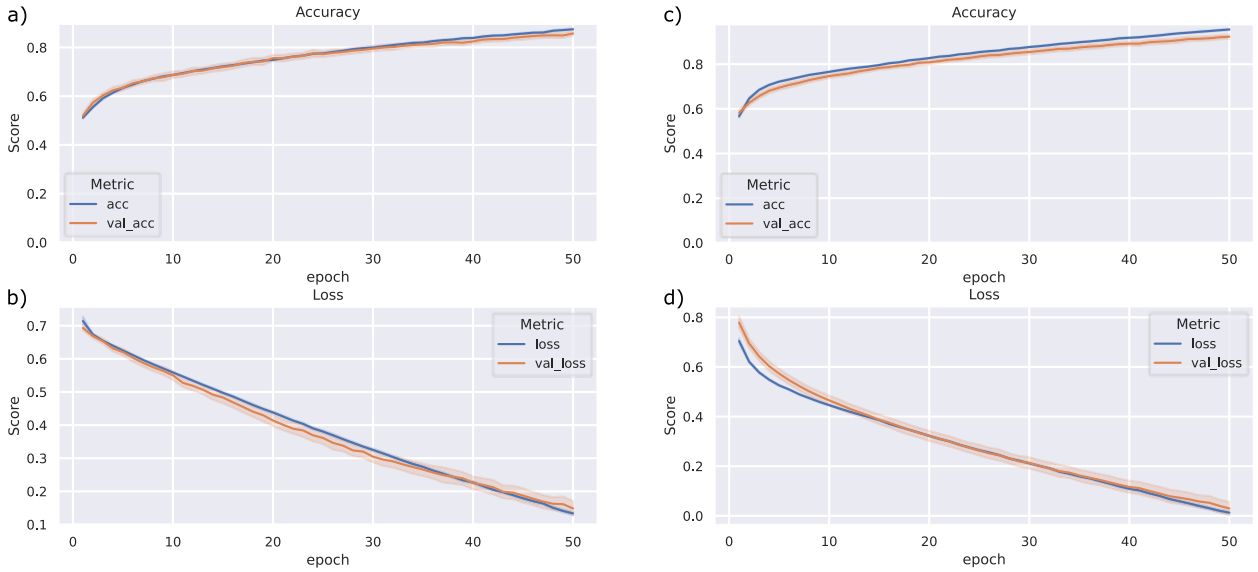

FIGURE 6. Training of the ResNet50 network with data augmentation by PCA as a function of epochs for training and validation data. Accuracy with the model trained from zeros a) and with learning transfer c). Loss with the model trained from zeros b) and with learning transfer d).

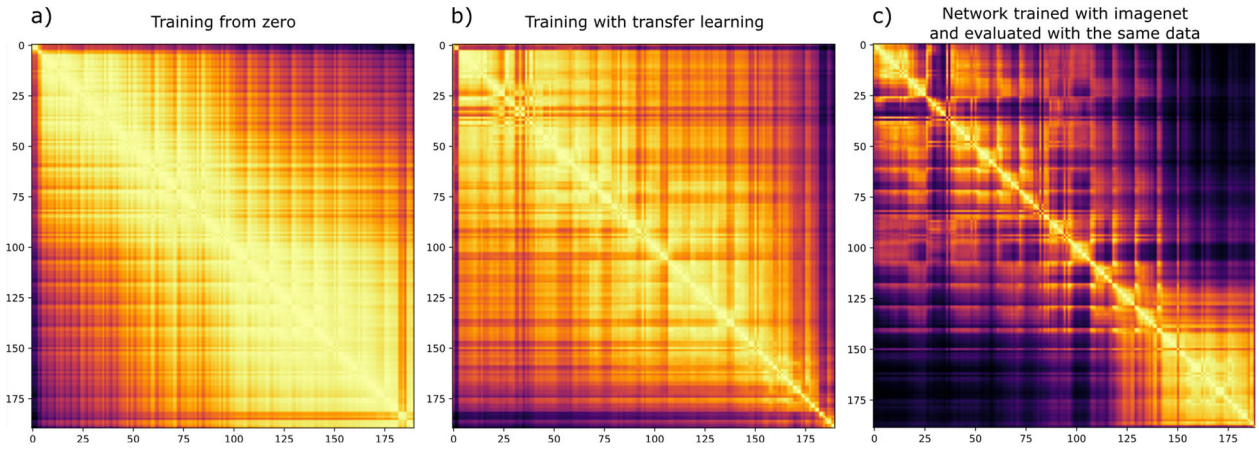

FIGURE 7. Similarity between layers given by the center kernel alignment coefficient. The similarity of the ResNet50 network a) trained with PCA data augmentation without learning transfer and evaluated with the MR images. b) trained with PCA data augmentation with learning transfer and evaluated with the MR images. c) trained with the ImageNet images and evaluated with the same images (reference network for learning transfer).

for the ResNet50 network with the PCA-based method are shown below. Additionally, the results with and without learning transfer are also included.

Figure 6 shows the training of the ResNet50 network for training with data augmentation by PCA, where Figure 6a and Figure 6b present the results starting from zeros and Figure 6c and Figure 6d with transfer learning. The results show similar behavior, i.e., progressive growth of model accuracy and decreasing losses as a function of epochs. Also, it is worth noting that the error bands are small, which implies a homogeneous training between the different model runs. On the other hand, the main difference between the training from zeros and the one implemented with the transfer learning lies in the fact that, for the first case, the network did not reach values as high as in the second case. In other words, transfer learning allowed for higher accuracy and reduced loss. In addition, the training and validation curves did not present significant differences, guaranteeing reduced overfitting, as can be deduced from the results obtained in Table 1.

Figure 7 shows the similarity between the 190 ResNet50 network layers for the case of training from zeros and with learning transfer. Additionally, Figure 7 presents the similarity between layers for the network trained with the ImageNet data, with the activation maps generated by the same dataset; in other words, Figure 7c is the reference similarity matrix. The reference matrix has little similarity between the farthest layers, i.e., between the first and the last layers. On the contrary, the closest layers present coefficients with similar values.

TABLE 2. p-value for the Kruskal Wallis test statistic between the different data augmentation methods generated results.

Training from zero	Based on PCA	Cropping	Color space	Translation	Flip	Distortion	Resizing	Linear filters	Rotation	Random frame deletion	Overlapping	Noise addition
Based on PCA	0,00	**0,09**	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Cropping	**0,09**	0,00	0,00	0,02	**0,31**	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Color space	0,00	0,00	0,00	0,01	0,00	**0,08**	**0,53**	**0,21**	0,01	**0,54**	0,00	0,00
Translation	0,00	0,02	0,01	0,00	**0,19**	**0,19**	0,02	0,00	**0,60**	0,00	0,00	0,00
Flip	0,01	**0,31**	0,00	**0,19**	0,00	0,01	0,00	0,00	**0,05**	0,00	0,00	0,00
Distortion	0,00	0,00	**0,08**	**0,19**	0,01	0,00	**0,24**	0,00	**0,34**	0,02	0,00	0,00
Resizing	0,00	0,00	**0,53**	0,02	0,00	**0,24**	0,00	**0,05**	0,05	**0,24**	0,00	0,00
Linear filters	0,00	0,00	**0,21**	0,00	0,00	0,00	**0,05**	0,00	0,00	**0,42**	0,00	0,00
Rotation	0,00	0,00	0,01	**0,60**	**0,05**	**0,34**	0,05	0,00	0,00	0,00	0,00	0,00
Random frame deletion	0,00	0,00	**0,54**	0,00	0,00	0,02	**0,24**	**0,42**	0,00	0,00	0,00	0,00
Overlapping	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,27**
Noise addition	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,27**	0,00

Training with transfer learning	Based on PCA	Flip	Distortion	Cropping	Color space	Translation	Random frame deletion	Resizing	Linear filters	Rotation	Overlapping	Noise addition
Based on PCA	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Flip	0,00	0,00	0,00	**0,32**	**0,96**	0,00	**0,08**	0,02	0,00	0,00	0,00	0,00
Distortion	0,00	0,00	0,00	0,00	0,00	0,00	**0,05**	0,00	0,00	0,00	0,00	0,00
Cropping	0,00	**0,32**	0,00	0,00	**0,22**	0,00	0,00	**0,17**	0,00	0,00	0,00	0,00
Color space	0,00	**0,96**	0,00	**0,22**	0,00	0,00	0,04	0,01	0,00	0,00	0,00	0,00
Translation	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	**0,13**	**0,61**	0,00	0,00
Random frame deletion	0,00	**0,08**	**0,05**	0,00	0,04	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Resizing	0,00	0,02	0,00	**0,17**	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Linear filters	0,00	0,00	0,00	0,00	0,00	**0,13**	0,00	0,00	0,00	0,03	0,00	0,00
Rotation	0,00	0,00	0,00	0,00	0,00	**0,61**	0,00	0,00	0,03	0,00	0,00	0,00
Overlapping	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Noise addition	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00

In the case of transfer learning training (Figure 7b), the pattern is preserved in the matrix; however, the similarity between layers increases in the layers far from each other. Finally, training from zeros (Figure 7a) essentially loses the pattern concerning the reference matrix; however, the layers at the extremes still have reduced similarity.

Table 2 and Table 3 show the p-value of the Kruskal Wallis test statistic. Table 2 shows the results between the data augmentation methods for the two cases: from zeros and with learning transfer. For the first case, it is observed that there is a more significant number of pairs of methods that have p-values above the significance level (greater than 0.05 highlighted in bold), indicating that the methods come from the same distribution, i.e., they have no difference between them. On the other hand, the proposed method only had a p-value above the significance level with the Cropping method with the network without learning transfer. Almost similarly, the proposed method did not have p-values above the significance level with any method in the case of the trained network with learning transfer, i.e., the proposed method is statistically different from the others.

Finally, Table 3 shows no p-value above the significance level, i.e., all methods have statistically significant differences when trained from zeros and with learning transfer, showing the high effectiveness of weight transfer.

IV. DISCUSSION

This paper presents a robust experimental framework for evaluating different data magnification methods in brain tumor detection with magnetic resonance imaging. The study was

TABLE 3. p-value for results trained from zero and implemented with transfer learning.

	p-value
Based on PCA	3,39E-21
Cropping	1,96E-19
Color space	1,05E-22
Translation	4,77E-16
Flip	1,32E-19
Distortion	5,64E-22
Resizing	6,54E-22
Linear filters	9,72E-17
Rotation	1,96E-19
Random frame deletion	4,35E-22
Overlapping	1,93E-20
Noise addition	9,62E-09

based on 12 different data augmentation methods, including a new image generation method based on principal component analysis (PCA). Additionally, a comparison between the training process from zero and with transfer learning is presented. The results showed the high effectiveness of the proposed method, achieving a maximum F1 score of 92.34% and outperforming the other evaluated methods. Additionally, all data augmentation methods were run in 40 runs to generate the distributions of model behaviors, with a better distribution observed for the proposed method under training with and without learning transfer. The scores of the distributions were subjected to the Kruskal Wallis non-parametric test statistic, where it was estimated that the proposed method is statistically different with a significance level of 0.05, guaranteeing the high effectiveness concerning the other conventional methods.

Although the results are promising, the work has some limitations or concepts that were not addressed in this article and would be interesting to explore as future work. For example, data augmentation was explored for each strategy individually; however, in some research, data augmentation is used by combining two or more strategies, creating a larger amount of data from the same reference image.

On the other hand, our focus was on 1.5 Tesla FLAIR images and, therefore, the results are extrapolated only to this type of image. Future work needs to be explored with other types of sequences, such as T2, T1, with contrast agents or proton density, and even with images generated by resonators of higher field strength (e.g., 7 Tesla). In this same sense, the study focused on the ResNet50 network since it is one of the most reported and efficient detection tasks. However, it is necessary to implement the strategies on other convolutional networks to generalize the results obtained in this work.

The proposed method showed promising results; however, the technique was used with a noise percentage of 90% on the principal components, being the noise ratio a variable that was not considered for the training of the models. In addition, since the images are subject to noise randomness, it is possible to generate several images from one. Therefore, it would be possible to explore the performance of the networks by augmenting the same image several times with this strategy. Finally, the study was performed on a single dataset, presenting homogeneity in the data, implying biased results towards that dataset.

V. CONCLUSION

An experimental framework for detecting brain tumors in magnetic resonance images was proposed, comparing 12 data augmentation methods with a new method based on principal component analysis. The generated images retained part of the spatial features, allowing to train the ResNet50 network until reaching an F1 score of 92.34%. The network, together with the proposed method, proved to be statistically different from conventional methods with a significance level of 0.05, guaranteeing the high effectiveness of the model. On the other hand, it was also possible to establish that data augmentation presents better results, generating significantly better models than models trained from zero.

APPENDIX

A. CONVOLUTIONAL NEURAL NETWORK RESNET50V2

The ResNet50v2 network consists mainly of convolutional layers, which use a convolutional operator. The operator, also known as filter or kernel, processes the image generating feature maps that are in turn used by the subsequent convolutional layers. The maps are patterns or abstractions that generally lack statistical inference but are the fundamental basis of the network to arrive at the desired task (e.g., detection) [83]. The ResNet50 network uses a total of 50 convolutional layers (see Figure 8); each map is established by the same mathematical model described by Equation (A.1).

$$A_j^{(l)} = \varphi^{(l)}\left(b_j^{(l)} + \sum_{i=1}^{M^{(l-1)}} A_i^{(l-1)} * K_{ij}^{(l)}\right) \tag{A.1}$$

here, $K_{ij}^{(l)}$ represents the j-th kernel of the l-th layer. $*$ is the convolutional operation between the kernel and the input feature map, which corresponds to the previous convolutional layer's output and has a depth of i feature maps. $b_j^{(l)}$ is the bias associated with the convolutional operation with the j-th kernel and $\varphi^{(l)}$ is the activation function of that layer [84], [85].

Additionally, the network is based on the concept of residual connection or mapping. In particular, the connection creates trajectories parallel to the convolutional layer sequences, allowing smooth transmission of the gradient through the layers and preventing the gradient value from being zero. Furthermore, the connection forces the network to learn the residual mapping $f(x) - x$, being easier to train if the ideal residual mapping is the identity function $f(x) = x$ (see Figure 8c) [86]. The convolutional layers are connected through such connections every three layers, as illustrated

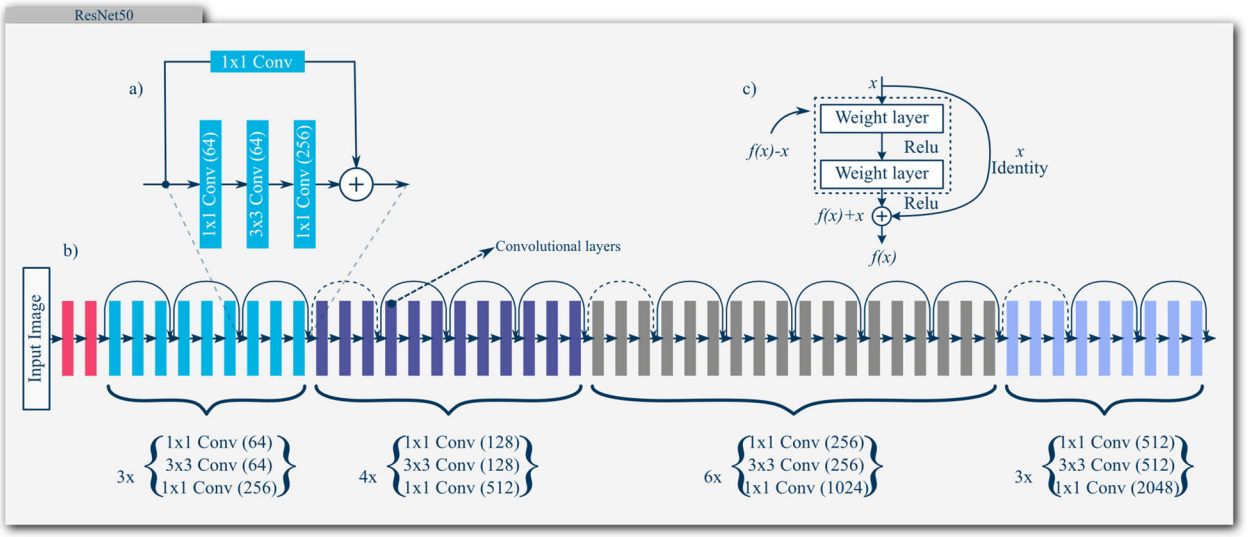

FIGURE 8. ResNet50 convolutional neural network. a) Block of three convolutional layers with the residual connection. b) General architecture of the ResNet50 network. c) Residual connection and mathematical model.

in Figure 8a and Figure 8b. Therefore, the output of each residual connection would be given by Equation (A.2) or its equivalent Equation (A.3), where R represents the output of the residual block.

$$A_j^{(R)} = \varphi^{(l)}\left(b_j^{(l)} + \sum_{i=1}^{M^{(l-1)}} A_i^{(l-1)} * K_{ij}^{(l)}\right) + \varphi^{(R)}\left(b_j^{(R)} + \sum_{i=1}^{M^{(l-3)}} A_i^{(l-3)} * K_{ij}^{(R)}\right) \quad \text{(A.2)}$$

$$A_j^{(R)} = \varphi^{(l)}\left(b_j^{(l)} + \sum_{i=1}^{M^{(l-1)}} A_i^{(l-1)} * K_{ij}^{(l)}\right) + \varphi^{(R)}\left(b_j^{(R)} + \sum_{i=1}^{M^{(R-1)}} A_i^{(R-1)} * K_{ij}^{(R)}\right) \quad \text{(A.3)}$$

On the other hand, although the ResNet50 network receives its name because it comprises 50 convolutional layers (see Figure 8.b), it has four types of layers apart from the convolutional layers, residual mapping input, and output. The additional layers are activation, pooling, batch normalization, and padding. In general, the ResNet50 consists of the 190 layers shown in Figure 9, being this the network implemented in this research. The additional layers are described in the following sections, except for the padding layer because it simply fills the images from zero to recover the original size lost after the convolutional layers.

1) ACTIVATION FUNCTION

In the mathematical models of the previous section, the activation function was defined and denoted by the Greek letter φ. The function is one of the fundamental elements in neural networks since it allows emulating the activation of the artificial neuron as a biological one would. The operation is constituted by a nonlinear relationship between the weighted input and the neuron's output and can vary depending on the design. However, we used the ReLu function [87] in this study since it allows faster training than other functions while maintaining its nonlinearity [88].

2) POOLING

In the convolution process, small changes on the input image generate small changes in the feature maps. Then, pooling layers were devised to endow the convolutional layers with some transitional invariance. Generally, the process calculates the maximum (or average, as the case may be) value for patches of a feature map and uses it to create a downsampled (clustered) feature map. In this sense, clustering reduces the size of feature maps, simplifies the model, and reduces the computational burden [89], [90].

3) BATCH NORMALIZATION

Batch normalization was devised to mitigate the problem of changing internal covariates produced by the change in the internal distribution of each feature map and the random initialization of the weights. The effect limits the learning rate but can be reduced by modifying the distribution toward a normal distribution, i.e., with mean 0 and standard deviation 1, as shown in Equation (19). The normalization is adjusted by training to an optimal distribution by a linear transformation, as shown in Equation (20). The parameters γ and β are learned by the model generating the new distribution, which improves the model performance [91]. The process also smooths the gradient flow and acts as a regularization layer [92]. Therefore, no additional regularization method

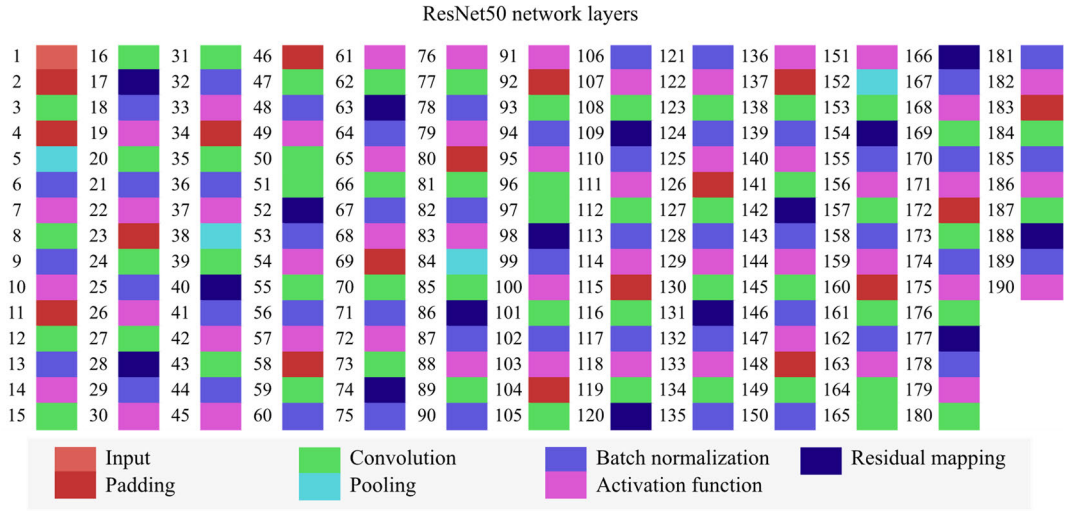

FIGURE 9. The complete structure of the resnet50 network.

was used in the implemented network.

$$A'^{(l)}_{Nj} = \frac{A^{(l)}_j - \mu^{(l)}_B}{\sqrt{\left(\sigma^{(l)}_B\right)^2 + \varepsilon}} \tag{19}$$

$$A^{(l)}_{Nj} = \gamma \cdot A'^{(l)}_{Nj} + \beta \tag{20}$$

In Equations (19) and (20), $A'^{(l)}_{Nj}$ represents the normalized feature map of the l-th layer, $A^{(l)}_{Nj}$ is the optimal distribution of the same layer, $A^{(l)}_j$ is the non-normalized input (see Equation (A.1)), $\mu^{(l)}_B$ and $\sigma^{(l)}_B$ represent the batch mean and variance respectively and ε is a stabilization coefficient, used to prevent the denominator from taking the value of 0.

B. DATA AUGMENTATION METHODS

This section shows how the algorithms of the main data augmentation methods work. In addition, the mathematical model governing each model is also included.

1) TRANSLATION

As mentioned above, the most common and simple methods are geometric transformations. The first of these is translation, which, as can be deduced from its name, the image is translated preserving the relative positions between pixels, but not its original position. Mathematically this operation is described by Equations (A.4) and (A.5).

$$x' = x + t_x \tag{A.4}$$

$$y' = x + t_y \tag{A.5}$$

where, x and y are the original positions of each pixel and x' and y' are the new positions resulting from the translation t_x and t_y. In some cases, Equations (A.4) and (A.5) are matrix represented as shown in Equation (A.6).

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{A.6}$$

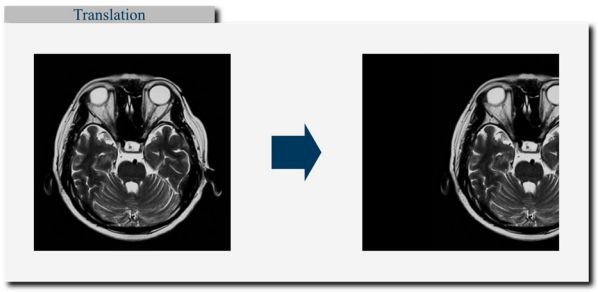

FIGURE 10. Horizontal translation.

here, the column vector is the translation vector. It is worth noting that, although Equation (A.6) is shown for two dimensions (on the x and y axis), the transformation can be applied for n dimensions, where the transformation vector will be a column vector of dimensions n [54], [55]. The above process is illustrated in Figure 10.

2) ROTATION

Rotation-based data augmentation is performed by rotating the image concerning its original position. Similar to translation, rotation consists of retaining the same relative position of the pixels but with a new coordinate axis system. Mathematically the transformation is given by Equation (A.7).

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{A.7}$$

here, again the rotation matrix is a square matrix where the angle θ is the rotation of the image concerning the origin [55], [56]. Rotations can take any angle, with rotations multiples of 90° being the most used in square images. In addition, rotations are generally taken concerning the image center and not from the origin, as illustrated in Figure 11.

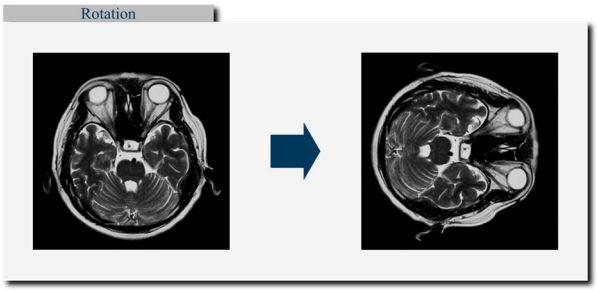

FIGURE 11. 90° clockwise rotation.

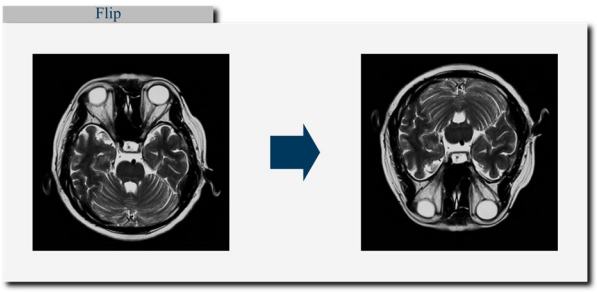

FIGURE 12. Vertical flip.

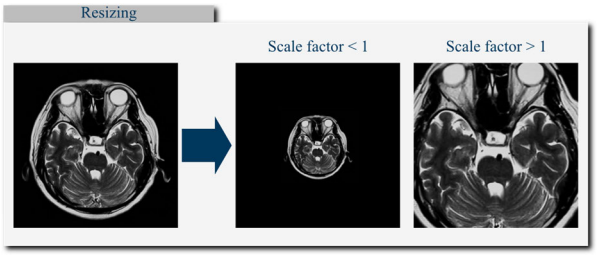

FIGURE 13. Example of image resizing with a scale factor less than 1 and one greater than 1.

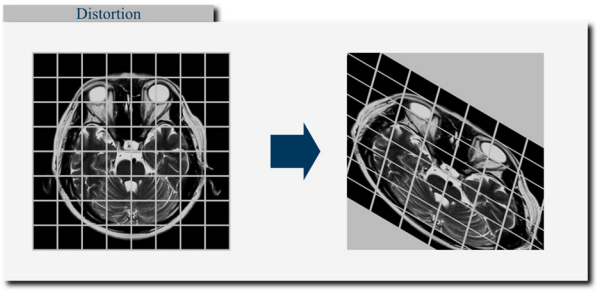

FIGURE 14. Distortion.

3) FLIP

Image flipping is another geometric transformation, where the position of the pixels is inverted concerning one of the two axes (in the case of two-dimensional data). The mathematical model is governed by Equation (A.8), and the process can be seen in Figure 12.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x_{max} \\ y_{max} \end{pmatrix} - \begin{pmatrix} x \\ y \end{pmatrix} \tag{A.8}$$

where, x_{max} and y_{max} are the last positions reached by the image pixels on the respective axes.

4) RESIZING

Resizing or rescaling consists of assigning the new positions in proportion to a scale factor, which may be the same for each axis or have different proportions. In particular, the change of scale can be interpreted as zoom in (scale factor >1) or zoom out (scale factor <1). Mathematically the resizing is expressed as shown in Equation (A.9).

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} f_x & 0 \\ 0 & f_y \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{A.9}$$

where, f_x, and f_y are the scale factors for the x and y axes, respectively. Figure 13 shows two examples of image resizing.

5) DISTORTION

Distortion shifts the position of pixels to new positions that follow some function. Even this strategy can be the combination of one or several translations, rotations, and resizing. For example, the distortion of Equation (A.10) contains the rotation, resizing, and translation processes in that respective order. It should be noted that Equation (A.10) represents the transformations in the homogeneous coordinates [57].

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} f_x \cdot \cos\theta & -f_x \cdot \sin\theta & t_x \\ f_y \cdot \sin\theta & f_y \cdot \cos\theta & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{A.10}$$

Figure 14 shows an example of distortion on an axial image of the brain.

Geometric transformations in discrete space (such as digital images) can generate new positions that do not correspond to an integer pixel. Consequently, transformations are used with interpolation methods to find intensity levels that correspond to discrete pixel positions. For example, Figure 15 shows the resizing of 2.5 on a 3×3 figure. The process would assign new positions to the initial pixels. However, these positions would not correspond to discrete positions, and, in addition, there would be intermediate pixels that would not have an assigned value. In this sense, interpolation becomes necessary to determine the intensity levels in the discrete positions and the intermediate pixels, being linear and cubic interpolation the most used [58]–[60].

6) CROPPING

Image blending is a rarely implemented strategy. The process involves taking elements from several images with the same features to generate a new image like a mosaic [19], [61]. For example, Figure 16 shows the composition of a new image from the regions of 9 different images with the same features (axial images).

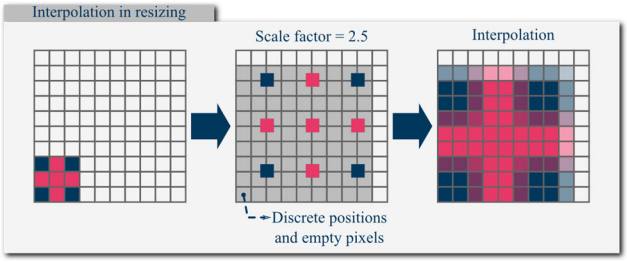

FIGURE 15. Representation of the resizing of a figure and interpolation.

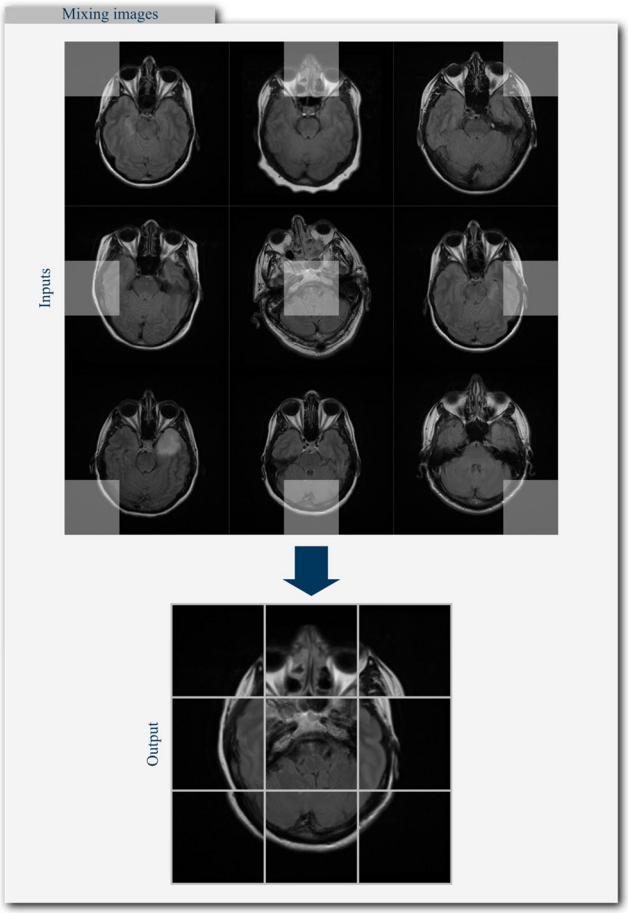

FIGURE 16. Mixing images with the same characteristics to generate a new image like a mosaic.

7) IMAGE OVERLAY

Another way of blending images is overlapping, as shown in Figure 17. The process consists of taking two images of the same size and matrix summing them multiplied by an attenuation factor [19].

8) NOISE INJECTION

Noise aggregation consists of summing a matrix of the same size with random values, usually with normal distributions (Gaussian) [62]. The process can help networks learn more robust functions by removing or hiding some image information, as illustrated in Figure 18 [19].

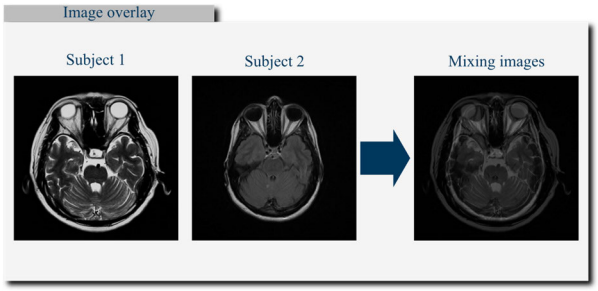

FIGURE 17. Superimposition of two images with the same characteristics.

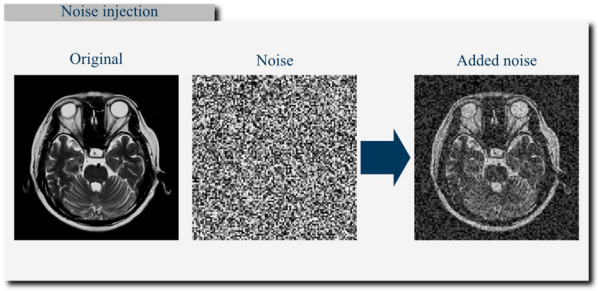

FIGURE 18. Random noise injection.

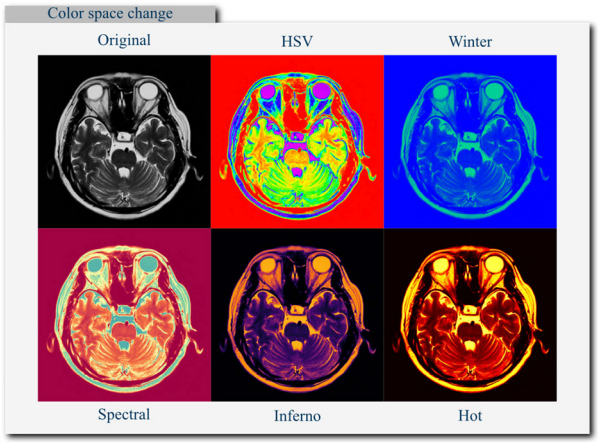

FIGURE 19. Changing the color space of a grayscale image to HSV, Winter, Spectral, Inferno, and Hot spaces.

9) COLOR SPACE

Generally, images are stored as arrays of three channels with the same dimensions. The channels represent each of the intensity levels that make up the RGB image, i.e., the intensities of red, green, and blue. Therefore, it is possible to change the image's color while preserving its spatial characteristics, as shown in Figure 19. The process is known as color space shift and can be performed in any number of spaces since each space combines the original channels in different proportions.

Color spaces can even be created by assigning to each intensity level a combination of the three RGB channels, i.e., a grayscale image (one channel) can be converted to an RGB space (three channels) [63]. The variety of color spaces is so

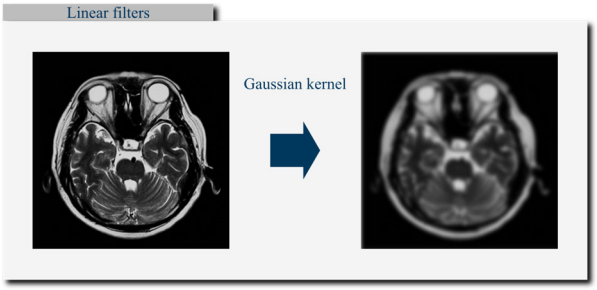

FIGURE 20. Blurring through a Gaussian filter.

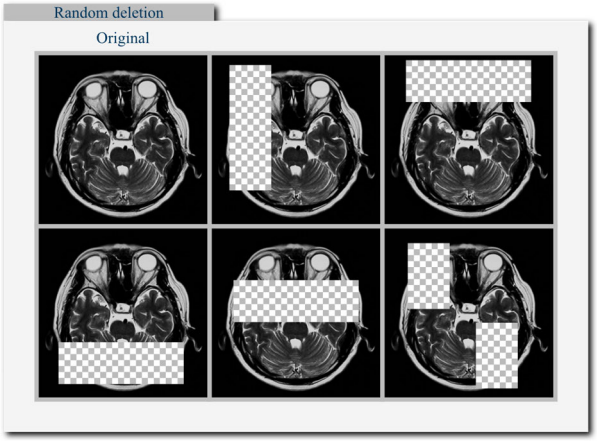

FIGURE 21. Random deletion of frames.

extensive that even PCA-based developments can be found, as performed by Krizhevsky *et al.* [93].

10) LINEAR FILTERS

Another of the most used strategies is linear filters. Generally, filters are used to focus and blur the image, as illustrated in Figure 20. The method consists in sliding the filter through the whole image, obtaining new values in the new image [64]. Particularly, this process is known as convolution and, in fact, is the fundamental basis in convolutional neural networks [65].

11) RANDOM DELETION OF FRAMES

Deletion is a strategy inspired by regularization based on neuron dropout. The process randomly eliminates regions of the image, preventing the neurons from learning part of the information, as illustrated in Figure 21 [66].

REFERENCES

[1] G. W. Brock, *The Second Information Revolution*. Cambridge, MA, USA: Harvard Univ. Press, 2021, doi: 10.4159/9780674028791.

[2] E. J. Topol, "High-performance medicine: The convergence of human and artificial intelligence," *Nature Med.*, vol. 25, no. 1, pp. 44–56, 2019, doi: 10.1038/s41591-018-0300-7.

[3] Z. Obermeyer and E. J. Emanuel, "Predicting the future—Big data, machine learning, and clinical medicine," *New England J. Med.*, vol. 375, no. 13, pp. 1216–1219, Sep. 2016, doi: 10.1056/NEJMp1606181.

[4] M. Haenlein and A. Kaplan, "A brief history of artificial intelligence: On the past, present, and future of artificial intelligence," *California Manage. Rev.*, vol. 61, no. 4, pp. 5–14, Aug. 2019, doi: 10.1177/0008125619864925.

[5] S. Legg and M. Hutter, "Universal intelligence: A definition of machine intelligence," *Minds Mach.*, vol. 17, no. 4, pp. 391–444, 2007, doi: 10.1007/s11023-007-9079-x.

[6] M. Rupali and P. Amit, "A review paper on general concepts of 'artificial intelligence and machine learning,'" *IARJSET*, vol. 4, no. 4, pp. 79–82, Jan. 2017, doi: 10.17148/IARJSET/NCIARCSE.2017.22.

[7] F. K. Dosilovic, M. Brcic, and N. Hlupic, "Explainable artificial intelligence: A survey," in *Proc. 41st Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, May 2018, pp. 210–215, doi: 10.23919/MIPRO.2018.8400040.

[8] A. Buetti-Dinh, V. Galli, S. Bellenberg, O. Ilie, M. Herold, S. Christel, M. Boretska, I. V. Pivkin, P. Wilmes, W. Sand, M. Vera, and M. Dopson, "Deep neural networks outperform human expert's capacity in characterizing bioleaching bacterial biofilm composition," *Biotechnol. Rep.*, vol. 22, Jun. 2019, Art. no. e00321, doi: 10.1016/j.btre.2019.e00321.

[9] C. Firestone, "Performance vs. competence in human–machine comparisons," *Proc. Nat. Acad. Sci. USA*, vol. 117, no. 43, pp. 26562–26571, Oct. 2020, doi: 10.1073/pnas.1905334117.

[10] W. Zhou, Y. Yang, C. Yu, J. Liu, X. Duan, Z. Weng, D. Chen, Q. Liang, Q. Fang, J. Zhou, H. Ju, Z. Luo, W. Guo, X. Ma, X. Xie, R. Wang, and L. Zhou, "Ensembled deep learning model outperforms human experts in diagnosing biliary atresia from sonographic gallbladder images," *Nature Commun.*, vol. 12, no. 1, p. 1259, Dec. 2021, doi: 10.1038/s41467-021-21466-z.

[11] D. Ravì, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G. Z. Yang, "Deep learning for health informatics," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 1, pp. 4–21, Jan. 2017, doi: 10.1109/JBHI.2016.2636665.

[12] F. Piccialli, V. D. Somma, F. Giampaolo, S. Cuomo, and G. Fortino, "A survey on deep learning in medicine: Why, how and when?" *Inf. Fusion*, vol. 66, pp. 111–137, Feb. 2021, doi: 10.1016/j.inffus.2020.09.006.

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, Feb. 2015, doi: 10.1038/nature14539.

[14] S.-H. Han, K. W. Kim, S. Kim, and Y. C. Youn, "Artificial neural network: Understanding the basic concepts without mathematics," *Dementia Neurocognitive Disorders*, vol. 17, no. 3, p. 83, 2018, doi: 10.12779/dnd.2018.17.3.83.

[15] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016, doi: 10.1016/j.neucom.2015.09.116.

[16] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, Apr. 2017, doi: 10.1016/j.neucom.2016.12.038.

[17] I. Castiglioni, L. Rundo, M. Codari, G. Di Leo, C. Salvatore, M. Interlenghi, F. Gallivanone, A. Cozzi, N. C. D'Amico, and F. Sardanelli, "AI applications to medical images: From machine learning to deep learning," *Phys. Medica*, vol. 83, pp. 9–24, Mar. 2021, doi: 10.1016/j.ejmp.2021.02.006.

[18] M. J. Willemink, W. A. Koszek, C. Hardell, J. Wu, D. Fleischmann, H. Harvey, L. R. Folio, R. M. Summers, D. L. Rubin, and M. P. Lungren, "Preparing medical imaging data for machine learning," *Radiology*, vol. 295, no. 1, pp. 4–15, Apr. 2020, doi: 10.1148/radiol.2020192224.

[19] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, p. 60, 2019, doi: 10.1186/s40537-019-0197-0.

[20] F. Pesapane, M. Codari, and F. Sardanelli, "Artificial intelligence in medical imaging: Threat or opportunity? Radiologists again at the forefront of innovation in medicine," *Eur. Radiol. Experim.*, vol. 2, no. 1, p. 35, Dec. 2018, doi: 10.1186/s41747-018-0061-6.

[21] J. C. Buckner, P. D. Brown, B. P. O'Neill, F. B. Meyer, C. J. Wetmore, and J. H. Uhm, "Central nervous system tumors," *Mayo Clinic Proc.*, vol. 82, no. 10, pp. 1271–1286, Oct. 2007, doi: 10.4065/82.10.1271.

[22] National Cancer Institute. (2021). *Risk Factors for Cancer*. Cancer.Gov. Accessed: Nov. 3, 2021. [Online]. Available: https://www.cancer.gov/about-cancer/causes-prevention/risk

[23] K. M. Vernau and P. J. Dickinson, "Brain tumors," *Consult. Feline Intern. Med.*, vol. 82, no. 5, pp. 505–516, 2006, doi: 10.1016/B0-72-160423-4/50057-3.

[24] J. Chen, W. Huan, H. Zuo, L. Zhao, C. Huang, X. Liu, S. Hou, J. Qi, and W. Shi, "Alu methylation serves as a biomarker for non-invasive diagnosis of glioma," *Oncotarget*, vol. 7, no. 18, pp. 26099–26106, May 2016, doi: 10.18632/oncotarget.8318.

[25] American Society of Clinical Oncology. (2021). *Brain Tumor: Statistics Cancer. Net Doctor-Approved Patient Information From ASCO*. ASCO. Accessed: Aug. 31, 2021. [Online]. Available: https://www.cancer.net/cancer-types/brain-tumor/statistics

[26] M. K. Abd-Ellah, A. I. Awad, A. A. M. Khalaf, and H. F. A. Hamed, "A review on brain tumor diagnosis from MRI images: Practical implications, key achievements, and lessons learned," *Magn. Reson. Imag.*, vol. 61, pp. 300–318, Sep. 2019, doi: 10.1016/j.mri.2019.05.028.

[27] C. M. L. Zegers, J. Posch, A. Traverso, D. Eekers, A. A. Postma, W. Backes, A. Dekker, and W. van Elmpt, "Current applications of deep-learning in neuro-oncological MRI," *Phys. Medica*, vol. 83, pp. 161–173, Mar. 2021, doi: 10.1016/j.ejmp.2021.03.003.

[28] A. B. Olin, C. Thomas, A. E. Hansen, J. H. Rasmussen, G. Krokos, T. G. Urbano, A. Michaelidou, B. Jakoby, C. N. Ladefoged, A. K. Berthelsen, K. Håkansson, I. R. Vogelius, L. Specht, S. F. Barrington, F. L. Andersen, and B. M. Fischer, "Robustness and generalizability of deep learning synthetic computed tomography for positron emission tomography/magnetic resonance imaging–based radiation therapy planning of patients with head and neck cancer," *Adv. Radiat. Oncol.*, vol. 6, no. 6, Nov. 2021, Art. no. 100762, doi: 10.1016/j.adro.2021.100762.

[29] G. Song, T. Shan, M. Bao, Y. Liu, Y. Zhao, and B. Chen, "Automatic brain tumour diagnostic method based on a back propagation neural network and an extended set-membership filter," *Comput. Methods Programs Biomed.*, vol. 208, Sep. 2021, Art. no. 106188, doi: 10.1016/j.cmpb.2021.106188.

[30] G. S. Tandel, A. Tiwari, and O. G. Kakde, "Performance optimisation of deep learning models using majority voting algorithm for brain tumour classification," *Comput. Biol. Med.*, vol. 135, Aug. 2021, Art. no. 104564, doi: 10.1016/j.compbiomed.2021.104564.

[31] M. O. Khairandish, M. Sharma, V. Jain, J. M. Chatterjee, and N. Z. Jhanjhi, "A hybrid CNN-SVM threshold segmentation approach for tumor detection and classification of MRI brain images," in *Proc. IRBM*, Jun. 2021, pp. 743–746, doi: 10.1016/j.irbm.2021.06.003.

[32] Z. A. Al-Saffar and T. Yildirim, "A hybrid approach based on multiple eigenvalues selection (MES) for the automated grading of a brain tumor using MRI," *Comput. Methods Programs Biomed.*, vol. 201, Apr. 2021, Art. no. 105945, doi: 10.1016/j.cmpb.2021.105945.

[33] A. S. M. Shafi, M. B. Rahman, T. Anwar, R. S. Halder, and H. M. E. Kays, "Classification of brain tumors and auto-immune disease using ensemble learning," *Informat. Med. Unlocked*, vol. 24, Jan. 2021, Art. no. 100608, doi: 10.1016/j.imu.2021.100608.

[34] C. J. Preetha, H. Meredig, G. Brugnara, M. A. Mahmutoglu, M. Foltyn, F. Isensee, T. Kessler, I. Pflüger, M. Schell, U. Neuberger, and J. Petersen, "Deep-learning-based synthesis of post-contrast T1-weighted MRI for tumour response assessment in neuro-oncology: A multicentre, retrospective cohort study," *Lancet Digit. Heal.*, vol. 3, no. 12, p. e784–e794, Oct. 2021, doi: 10.1016/S2589-7500(21)00205-3.

[35] S. Amemiya, H. Takao, S. Kato, H. Yamashita, N. Sakamoto, and O. Abe, "Automatic detection of brain metastases on contrast-enhanced CT with deep-learning feature-fused single-shot detectors," *Eur. J. Radiol.*, vol. 136, Mar. 2021, Art. no. 109577, doi: 10.1016/j.ejrad.2021.109577.

[36] J. Yan *et al.*, "Deep learning features from diffusion tensor imaging improve glioma stratification and identify risk groups with distinct molecular pathway activities," *eBioMedicine*, vol. 72, Oct. 2021, Art. no. 103583, doi: 10.1016/j.ebiom.2021.103583.

[37] M. Islam, N. Wijethilake, and H. Ren, "Glioblastoma multiforme prognosis: MRI missing modality generation, segmentation and radiogenomic survival prediction," *Computerized Med. Imag. Graph.*, vol. 91, Jul. 2021, Art. no. 101906, doi: 10.1016/j.compmedimag.2021.101906.

[38] B. Menze, F. Isensee, R. Wiest, B. Wiestler, K. Maier-Hein, M. Reyes, and S. Bakas, "Analyzing magnetic resonance imaging data from glioma patients using deep learning," *Computerized Med. Imag. Graph.*, vol. 88, Mar. 2021, Art. no. 101828, doi: 10.1016/j.compmedimag.2020.101828.

[39] M. Jiang, F. Zhai, and J. Kong, "A novel deep learning model DDU-Net using edge features to enhance brain tumor segmentation on MR images," *Artif. Intell. Med.*, vol. 121, Nov. 2021, Art. no. 102180, doi: 10.1016/j.artmed.2021.102180.

[40] M. Decuyper, S. Bonte, K. Deblaere, and R. Van Holen, "Automated MRI based pipeline for segmentation and prediction of grade, IDH mutation and 1p19q co-deletion in glioma," *Computerized Med. Imag. Graph.*, vol. 88, Mar. 2021, Art. no. 101831, doi: 10.1016/j.compmedimag.2020.101831.

[41] Y. L. Wang, Z. J. Zhao, S. Y. Hu, and F. L. Chang, "CLCU-Net: Cross-level connected U-shaped network with selective feature aggregation attention module for brain tumor segmentation," *Comput. Methods Programs Biomed.*, vol. 207, Aug. 2021, Art. no. 106154, doi: 10.1016/j.cmpb.2021.106154.

[42] A. Khosravanian, M. Rahmanimanesh, P. Keshavarzi, and S. Mozaffari, "Fast level set method for glioma brain tumor segmentation based on superpixel fuzzy clustering and lattice Boltzmann method," *Comput. Methods Programs Biomed.*, vol. 198, Jan. 2021, Art. no. 105809, doi: 10.1016/j.cmpb.2020.105809.

[43] R. Poel, E. Rüfenacht, E. Hermann, S. Scheib, P. Manser, D. M. Aebersold, and M. Reyes, "The predictive value of segmentation metrics on dosimetry in organs at risk of the brain," *Med. Image Anal.*, vol. 73, Oct. 2021, Art. no. 102161, doi: 10.1016/j.media.2021.102161.

[44] K. K. Wong, J. S. Cummock, Y. He, R. Ghosh, J. J. Volpi, and S. T. C. Wong, "Retrospective study of deep learning to reduce noise in non-contrast head CT images," *Computerized Med. Imag. Graph.*, vol. 94, Dec. 2021, Art. no. 101996, doi: 10.1016/j.compmedimag.2021.101996.

[45] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle, and L. Tarbox, "The cancer imaging archive (TCIA): Maintaining and operating a public information repository," *J. Digit. Imag.*, vol. 26, no. 6, pp. 1045–1057, Dec. 2013, doi: 10.1007/s10278-013-9622-7.

[46] N. Pedano *et al.*, "Radiology data from the cancer genome atlas low grade glioma [TCGA-LGG] collection," Cancer Imag. Arch., 2016, doi: 10.7937/K9/TCIA.2016.L4LTD3TK.

[47] J.-G. Lee, S. Jun, Y.-W. Cho, H. Lee, G. B. Kim, J. B. Seo, and N. Kim, "Deep learning in medical imaging: General overview," *Korean J. Radiol.*, vol. 18, no. 4, p. 570, 2017, doi: 10.3348/kjr.2017.18.4.570.

[48] A. S. Lundervold and A. Lundervold, "An overview of deep learning in medical imaging focusing on MRI," *Zeitschrift Für Medizinische Physik*, vol. 29, no. 2, pp. 102–127, 2019, doi: 10.1016/j.zemedi.2018.11.002.

[49] B. J. Erickson, "Basic artificial intelligence techniques," *Radiolog. Clinics North Amer.*, vol. 59, no. 6, pp. 933–940, Nov. 2021, doi: 10.1016/j.rcl.2021.06.004.

[50] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artif. Intell. Rev.*, vol. 53, pp. 5455–5516, Apr. 2020, doi: 10.1007/s10462-020-09825-6.

[51] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[53] P. Chlap, H. Min, N. Vandenberg, J. Dowling, L. Holloway, and A. Haworth, "A review of medical image data augmentation techniques for deep learning applications," *J. Med. Imag. Radiat. Oncol.*, vol. 65, no. 5, pp. 545–563, Aug. 2021, doi: 10.1111/1754-9485.13261.

[54] A. C. Bovik, "Basic gray-level image processing," in *Handbook of Image and Video Processing*. Amsterdam, The Netherlands: Elsevier, 2005, pp. 21–37.

[55] B. Zoph, E. D. Cubuk, G. Ghiasi, T. Y. Lin, J. Shlens, and Q. V. Le, "Learning data augmentation strategies for object detection," in *Computer Vision—ECCV 2020* (Lecture Notes in Computer Science), vol. 12372, A. Vedaldi, H. Bischof, T. Brox, and J. M. Frahm, Eds. Cham, Switzerland: Springer, 2020, doi: 10.1007/978-3-030-58583-9_34.

[56] H. Malepati, "Advanced image processing algorithms," in *Digital Media Processing*. Amsterdam, The Netherlands: Elsevier, 2010, pp. 553–592.

[57] F. Brill, V. Erukhimov, R. Giduthuri, and S. Ramm, "Basic image transformations," in *OpenVX Programming Guide*. Amsterdam, The Netherlands: Elsevier, 2020, pp. 85–123.

[58] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 6, pp. 1153–1160, Dec. 1981, doi: 10.1109/TASSP.1981.1163711.

[59] N. A. Dodgson, "Quadratic interpolation for image resampling," *IEEE Trans. Image Process.*, vol. 6, no. 9, pp. 1322–1326, Sep. 1997, doi: 10.1109/83.623195.

[60] A. M. Bayen and T. Siauw, "Interpolation," in *An Introduction to MATLAB Programming and Numerical Methods for Engineers*. Amsterdam, The Netherlands: Elsevier, 2015, pp. 211–223.

[61] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep CNNs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 9, pp. 2917–2931, Sep. 2020, doi: 10.1109/TCSVT.2019.2935128.

[62] T. DeVries and G. W. Taylor, "Dataset augmentation in feature space," 2017, *arXiv:1702.05538*.
[63] D. R. Bull and F. Zhang, "Color spaces and color transformations," in *Intelligent Image and Video Compression*. Amsterdam, The Netherlands: Elsevier, 2021, pp. 107–142.
[64] L. Tan and J. Jiang, "Image filtering enhancement," in *Digital Signal Processing*. Amsterdam, The Netherlands: Elsevier, 2019, pp. 649–726.
[65] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018, doi: 10.1016/j.patcog.2017.10.013.
[66] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proc. IEEE AAAI Conf. Artif. Intell.*, vol. 34, no. 7, May 2020, pp. 13001–13008, doi: 10.1609/aaai.v34i07.7000.
[67] A. M. Jade, B. Srikanth, V. K. Jayaraman, B. D. Kulkarni, J. P. Jog, and L. Priya, "Feature extraction and denoising using kernel PCA," *Chem. Eng. Sci.*, vol. 58, no. 19, pp. 4441–4448, Oct. 2003, doi: 10.1016/S0009-2509(03)00340-3.
[68] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jan. 1991, pp. 586–591, doi: 10.1109/CVPR.1991.139758.
[69] J. H. Friedman, "Exploratory projection pursuit," *J. Amer. Statist. Assoc.*, vol. 82, no. 397, pp. 249–266, Mar. 1987, doi: 10.1080/01621459.1987.10478427.
[70] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.
[71] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: A survey," *Knowl.-Based Syst.*, vol. 80, pp. 14–23, May 2015, doi: 10.1016/j.knosys.2015.01.010.
[72] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," 2014, *arXiv:1409.0575*.
[73] V. Andreieva and N. Shvai, "Generalization of cross-entropy loss function for image classification," *Mohyla Math. J.*, vol. 3, pp. 3–10, Jan. 2021, doi: 10.18523/2617-7080320203-10.
[74] Y.-D. Ma, Q. Liu, and Z.-B. Quan, "Automated image segmentation using improved PCNN model based on cross-entropy," in *Proc. Int. Symp. Intell. Multimedia, Video Speech Process.*, Oct. 2004, pp. 743–746, doi: 10.1109/ISIMP.2004.1434171.
[75] A.-M. Šimundić, "Measures of diagnostic accuracy: Basic definitions," *EJIFCC*, vol. 19, no. 4, pp. 203–211, Jan. 2009. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/27683318
[76] R. Trevethan, "Sensitivity, specificity, and predictive values: Foundations, pliabilities, and pitfalls in research and practice," *Frontiers Public Health*, vol. 5, p. 307, Nov. 2017, doi: 10.3389/fpubh.2017.00307.
[77] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," 2020, *arXiv:2010.16061*.
[78] A. F. M. Alkarkhi, "The observed significance level (P-value) procedure," in *pplications of Hypothesis Testing for Environmental Science*. Amsterdam, The Netherlands: Elsevier, 2021, pp. 79–119, doi: 10.1016/B978-0-12-824301-5.00010-1.
[79] P. E. McKight and J. Najab, "Kruskal–Wallis test," in *The Corsini Encyclopedia Psychology*, vol. 30. Hoboken, NJ, USA: Wiley, 2010, doi: 10.1002/9780470479216.corpsy0491.
[80] E. Ostertagová, O. Ostertag, and J. Kováč, "Methodology and application of the Kruskal–Wallis test," *Appl. Mech. Mater.*, vol. 611, pp. 115–120, Aug. 2014, doi: 10.4028/www.scientific.net/AMM.611.115.
[81] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, "Similarity of neural network representations revisited," 2019, *arXiv:1905.00414*.
[82] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf, "Measuring statistical dependence with Hilbert-Schmidt norms," in *Algorithmic Learning Theory* (Lecture Notes in Computer Science), vol. 3734, S. Jain, H. U. Simon, and E. Tomita, Eds. Berlin, Germany: Springer, 2005, doi: 10.1007/11564089_7.
[83] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *Proc. Int. Conf. Eng. Technol. (ICET)*, Aug. 2017, pp. 1–6, doi: 10.1109/ICEngTechnol.2017.8308186.
[84] C.-C. J. Kuo, "Understanding convolutional neural networks with a mathematical model," *J. Vis. Commun. Image Represent.*, vol. 41, pp. 406–413, Nov. 2016, doi: 10.1016/j.jvcir.2016.11.003.
[85] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, "A guide to convolutional neural networks for computer vision," *Synth. Lectures Comput. Vis.*, vol. 8, no. 1, pp. 1–207, 2018, doi: 10.2200/S00822ED1V01Y201712COV015.
[86] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision–ECCV 2016*. Cham, Switzerland: Springer, 2016, pp. 630–645, doi: 10.1007/978-3-319-46493-0_38.
[87] V. Nair and G. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814. [Online]. Available: https://icml.cc/Conferences/2010/papers/432.pdf
[88] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9. [Online]. Available: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf
[89] M. Ranzato, F. J. Huang, Y.-L. Boureau, and Y. LeCun, "Unsupervised learning of invariant feature hierarchies with applications to object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8, doi: 10.1109/CVPR.2007.383157.
[90] A. Giusti, D. C. Ciresan, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Fast image scanning with deep max-pooling convolutional neural networks," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 4034–4038, doi: 10.1109/ICIP.2013.6738831.
[91] I. Goodfellow, Y. Bengio, and A. Courville, "Optimization for training deep models," in *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016, pp. 313–317. [Online]. Available: http://www.deeplearningbook.org.
[92] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, vol. 1, Feb. 2015, pp. 448–456.
[93] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.