2023 IEEE Latin American Conference on Computational Intelligence (LA-CCI) | 979-8-3503-4807-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/LA-CCI58595.2023.10409397

Regularizing Neural Networks with Noise Injection for Classification of Brain Tumor in Magnetic Resonance Imaging

Umberto Tenório de Barros Filho
Unicap-Icam International School
Universidade Católica de Pernambuco
Recife, Brazil
umberto.2017205685@unicap.br

Paulo Rocha
Department of Internal Medicine
University of California, Davis
Sacramento, US
phrocha@ucdavis.edu

Marcos Oliveira
Department of Computer Science
University of Exeter
United Kingdom
M.A.Oliveira@exeter.ac.uk

Andrea Maria Nogueira Cavalcanti Ribeiro
Unicap-Icam International School
Universidade Católica de Pernambuco
Recife, Brazil
andrea.ribeiro@unicap.br

Rodrigo de Paula Monteiro
Unicap-Icam International School
Universidade Católica de Pernambuco
Recife, Brazil
rodrigo.paula@unicap.br

Diego Pinheiro
Universidade de Pernambuco
Recife, Brazil
dmpfs@ecomp.poli.br

***Abstract*—Overfitting jeopardizes deep learners dealing with noisy medical data. Despite its importance, our understanding of how neural networks generalize in medical imaging remains elusive. In this work, we propose to characterize the *generalization profile* of a model by evaluating its F1-score under data with varying noise levels. We use two architecturally distinct neural models, a Multilayer Perceptron (MLP) and a Convolutional Neural Network (CNN), to classify brain tumor in Magnetic Resonance Images (MRI) in a independent test data with different levels of salt-and-pepper noise. Our results reveal a clear distinction between the generalization profile of MLPs and CNNs. We demonstrate that shallow models can perform comparably to deep models when regularization is neglected. We also show that deep neural models benefit more from noise injection than shallow neural models, regardless of the differences between training and test data distribution. Our study sheds light on the nature of generalization in neural networks, laying the groundwork for further investigations of their underlying learning processes.**

***Index Terms*—overffiting, regularizatoin, generalization, classification, noise injection, neural networks**

I. Introduction

Overfitting is an increasing challenge as models become more complex and learn from noisy training data. The phenomenon occurs when a model fits in-sample data more than it is warranted to generalize to out-of-sample data [1]–[3]. These over-fitted models, instead of learning the underlying patterns, memorize the training data and its inherent noise [1]–[3]. While such models may perform well on the training set, their performance on the test set is compromised due to their limited ability to adapt to differences between the training and test data. Both shallow and deep learning models, such as Multilayer Perceptron (MLP) and Convolutional Neural Networks (CNN), often struggle with overfitting and, as a result, require regularization such as batch normalization, dropout techniques, and data augmentation [2].

While regularization help neural network learning, handling noisy data, especially in medical imaging, remains challenging. In the field, neural networks learn patterns from medical images [4]–[6] acquired from computed tomography, x-ray, ultrasound, and MRIs [4]–[6]. Techniques which often result in issues including low Signal-to-Noise Ratio (SNR) and low Contrast-to-Noise Ratio (CNR) [4], [7]. Such noise, however, induces biases that distance training and test data. Yet we have only a limited understanding of regularization methods for complex neural networks learning from noisy medical imaging [8].

In this context, Noise Injection (NI) is a sophisticated data augmentation regularization that can effectively reduce overfitting of neural networks learning from in medical imaging [9]. For instance, in ultrasound data, NI has proven more effective than other regularization strategies such as weight decay and early stopping [9], [10]. NI regularization augments data by injecting noise into training data and thus prevent models to learn Gaussian Noise [11]. Most NI regularization focus on Gaussian Noise and other types of noise such as Salt & Pepper are overlooked [12].

The contribution of this work is twofold. First, we employ NI to characterize the generalization profile of shallow and deep neural network models, specifically MLP and CNN, under training and test data distributions that incorporate varying levels of salt-and-pepper noise. Second, we use NI regularization to augment the data with different levels of salt-

979-8-3503-4807-1/23/$31.00 ©2023 IEEE

and-pepper noise. Our research underscore that distinct neural network models present unique generalization profile when evaluated with test data other than those identically distributed to training data. Additionally, we show that regularization improve model's capacity to generalize to out-of-sample data other than those identically distributed to in-sample data.

The subsequent parts of this paper are structured as follows. Section II details the data, noise datasets generation, neural networks architecture and evaluation metrics used in this study. Section III showcases the plots obtained from f-score analysis. Our results and most relevant findings are discussed in Section IV. The paper concludes with a summary of the paper and future avenues for research in Section V. Overall, our work offers valuable insights about the process of generalization in neural networks, paving the way for better understanding their fundamental learning processes.

II. Methods

The Methods section is organised as follows. Commencing with Subsection II-A detailing the 3,000 brain tumor MRI dataset acquired from two distinct sources [13], [14]. In Subsection II-B, this same dataset was replicated and subsequently subjected to different levels of Salt & Pepper noise from 10% to 90%, at intervals of 10%. Subsection II-C introduces two neural models used in this study, a shallow MLP and a deep CNN. Then, in Subsection II-D, a 5-fold cross-validation is shown for model training and testing in two scenarios: training and testing only with a specific noise dataset and training with every noise dataset combined and testing in each noise level. Finally, Subsection II-E explores f1-score metric obtained from the mean of each 5-fold result, providing a comprehensive assessment of model performance.

A. Dataset of Magnetic Resonance Images

The dataset used to train and test the models was assembled from two sources and consists of a total of $3,000$ brain tumor MRI images. The images were obtained from publicly accessible databases on the Kaggle [13] and FigShare [14]. These images were categorized into two classes: suggestive of cancer and non-suggestive of cancer (Figure 1). The suggestive category consists of images showing different types and stages of brain tumors, while the non-suggestive category consists of images with no evident signs of tumors.

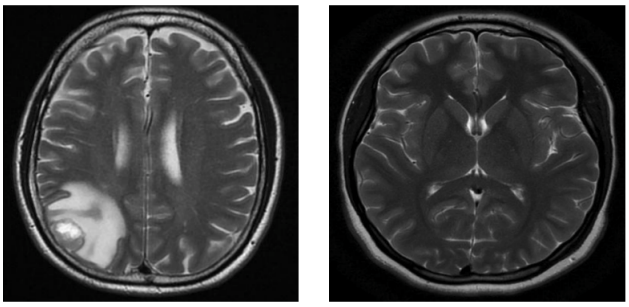

Fig. 1. Examples of suggestive (left) and non-suggestive (right) cancer images.

The image dataset, before training the model, was subjected to a data augmentation process which generated new images by introducing transformations into original images. The augmentation techniques included rotation up to 40 degrees as well as horizontal and vertical flipping. Images have different dimensions, ranging from 200×200 pixels to $1,200 \times 1,200$ pixels. Images were resized to a uniform dimension of 200×200 pixels. Images, although essentially gray-scale, were originally encoded in the Red-Green-Blue (RGB) color space but were converted to gray-scale.

All of the code, datasets, and analysis are available on the Open Science Framework (OSF) repository of this project at https://doi.org/10.17605/OSF.IO/UKCBX.

B. Generation of Noise in MRI Images

Nine noisy datasets were created by introducing noise into the original dataset. The noise levels were set at intervals of 10% ranging from 10% to 90% (Figure 2). This allowed the models to be trained and tested on identical images corrupted by varying levels of noise. The noise applied to the images was in the form of Salt & Pepper noise. This is a type of image noise that randomly replaces a certain percentage of image pixels with either black or white pixels. The noise level determines the probability each pixel is replaced. By following this methodology, the models were trained and tested on the same set of images with all combinations noise levels.

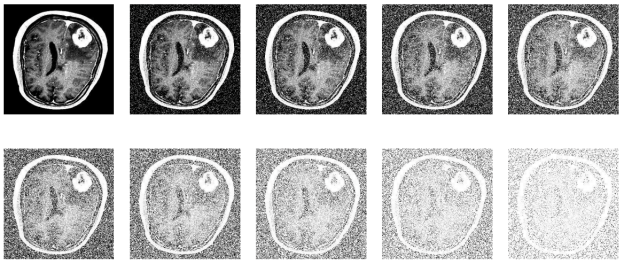

Fig. 2. MRI images with Salt & Pepper noise levels from 0% to 90% (i,e., top-left to bottom-right) applied.

C. Neural Network Models

Two neural network models with distinct architectures but similar numbers of parameters were compared in this study. The first model is a shallow MLP with approximately 15 million parameters, referring to the weights and biases that define the connections and activations of the MLP's layers. The second model is a deep CNN [12] with approximately 20 million parameters, including convolutional filters, pooling sizes, and fully connected layers' weights and biases.

The deep CNN architecture, related to [7] (Table I), incorporates two convolutional layers with 32 and 64 filters, respectively, each with a kernel size of 5×5 pixels. These layers are interleaved by two max-pooling layers, each with pool size of 2×2 pixels. Next, a flatten layer reshapes the multidimensional convolutional output into a one-dimensional vector. This vector is then relayed through a fully connected Dense layer with 128 neurons, followed by a dropout layer to

help reduce overfitting. The architecture concludes with a fully connected output layer that performs binary classification.

TABLE I
CNN ARCHITECTURE.

Layer	Output Shape	Params
Convolutional	$200 \times 200 \times 32$	2432
Max Pooling	$100 \times 100 \times 32$	0
Convolutional	$100 \times 100 \times 64$	51264
Max Pooling	$50 \times 50 \times 64$	0
Flatten	160000	0
Dense	128	20480128
Dropout	128	0
Dense	1	129

A shallow MLP counterpart was implemented to maintain a comparable number of parameters with the deep CNN. Its architecture begins with a flatten layer that transforms the two-dimensional image data into a one-dimensional vector. This vector is then transmitted through four fully-connected dense layers, each containing 128, 128, 64, and 64 neurons, respectively. Like the CNN, it concludes with a fully Connected output layer that performs binary classification (Table II).

The relu was the activation function of internal layers for both CNN and MLP. The sigmoid was the activation function of the final output layer for both CNN and MLP. All other necessary model parameters were set to the default values provided by the Keras library [15].

TABLE II
MLP ARCHITECTURE.

Layer	Output Shape	Params
Flatten	120000	0
Dense	128	15360128
Dense	128	16512
Dense	64	8256
Dense	64	4160
Dense	1	65

D. Model Training and Testing

Each model, shallow MLP and deep CNN, was separately trained for each noise level introduced to the dataset. Despite being trained on a specific noise level, each trained model was also separately tested on each noise level (Figure 3).

The training dataset comprised 80% of the total images ($2,400$ images), with the remaining 20% (600 images) reserved for testing. It is worth mentioning that 10% (240 images) of the training dataset was reserved for validation during each training epoch. This strategy served as an additional measure to avoid overfitting in both the MLP and CNN models.

Additionally, another analysis was conducted by concatenating the individual noise levels replicated training datasets ($2,400$ images) to construct a training scenario with $24,000$ images containing all noise levels. Both models were trained in this training scenario and then tested on each test noise level (Figure 4). The purpose of this analysis was to understand the models behavior and performance in handling different levels of noise in the training data.

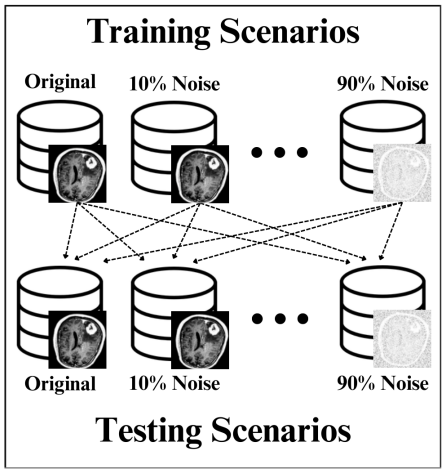

Fig. 3. The original dataset is replicated for each noise level in this study. 80% of the data ($2,400$ images) was used for training, while 20% (600 images) is reserved for testing. Both models were trained and tested on each noise level to evaluate their generalization capabilities.

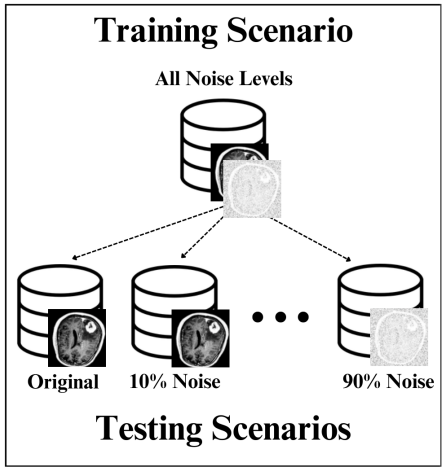

Fig. 4. In the second analysis, the training dataset is created by concatenating all noise levels replicas, resulting in a set of 24,000 images. The models are trained using this dataset and subsequently tested on each individual noise level to better understand the models performance and generalization capabilities when handling different levels of noise.

To ensure a reliable evaluation of the models' performance, a 5-fold cross-validation technique was employed to partition the image set into five equally-sized subsets [5]. At each partition, four subsets were used for training and the remaining one for testing. This process was repeated such that each subset served as the test set once. The overall performance of the models was then determined by averaging the validation results obtained from the 5-folds.

E. Characterization of Generalization Profile

Among all available model performance metrics, the f1-score metric was selected as the primary metric for this study analysis. The f1-score effectively demonstrates the performance of a model through the harmonic mean between two competing metrics: recall and precision, as seen in (1).

$$fscore = \frac{2}{recall^{-1} + precision^{-1}} \tag{1}$$

In the context of binary classification, recall, also known as sensitivity or true positive rate, is the fraction of actual positive instances that are correctly classified as positive by the model. Precision, also known as positive predictive value, is the proportion of correctly classified instances out of all instances that the model has predicted for a particular class.

The classification report yielded four distinct f1-scores: one for positive instances, one for negative instances, one as the average of the previous two, and one as a weighted (or balanced) average, which takes into account the relative size of each class. As the number of images in the dataset is balanced across classes, there was no difference between the weighted and unweighted f1-score averages.

III. Results

The Results section unfolds across three enlightening subsections. Subsection III-A introduces the heatmap, demonstrating model outcomes in the noise scenarios (ranging from 10% to 90% noise levels for both training and testing). Subsection III-B presents a confidence intervals per heatmap row, showcasing models' in-sample and out-of-sample behavior, revealing the overfitting behavior. Subsection III-C demonstrates models trained with combined noise levels, unveiling enhanced stability and performance when tested in every noise case.

A. Distinct Generalization Profile

A total of 100 f1-scores were obtained for each type of neural architectures. The f1-score results for the CNN and MLP models are presented as heatmaps, with each cell showing the average f1-score calculate over the 5-fold cross-validation under each specific training and testing scenario (Figure 5). The horizontal and vertical axes displaying varying noise levels introduced to the test and training data, respectively. Warmer colors in the heatmap denote higher scores, signifying superior model performance, whereas cooler colors suggest lower scores, indicating a relative decline in performance.

The trace (i.e., diagonal), upper triangle and lower triangle of heatmaps, indicate scenarios in which models were tested with noise levels similar, greater, and lower, respectively, to noise levels in which they were trained. For CNN (Figure 5, left), such values are 0.89, 0.69, and 0.70, respectively. For MLP (Figure 5, right), the corresponding mean f1-scores are: 0.75, 0.53, and 0.74.

B. Overfitting Impact In-Sample and Out-of-Sample

The mean of each training scenario was measured (i.e., the average of each row in the heatmap of Figure 5) and the 95% confidence intervals were obtained for both models (Figure 6). In the CNN, the f1-score interval are larger for lower training noise levels and smaller for higher training noise levels. A different behavior occurs in the MLP, the f1-score intervals are smaller in almost every noise scenario, with the best mean f1-score for higher training noise.

Overfitting becomes apparent as the F1-score declines when out-of-sample noise deviates from the in-sample noise distribution. This shows the model's excessive adaptation to noise within the training data, resulting in limited generalization to new instances. By the way, the model's adaptation to noise also exhibits a narrow interval, underscoring its stability in comprehending noise.

C. Noise Injection Data Augmentation

Considering the second analysis, which involved training with $24,000$ images across various noise levels (Figure 4), the CNN model consistently achieved higher mean values of f1-score for all testing noise levels (Figure 7). The mean standard error remained constant for both models, as did the difference between the mean scores of both models across all noise levels.

IV. Discussion

Our understanding of how neural networks generalize in medical imaging remains elusive. This study demonstrated the improvements related to overfitting that noise injection can bring by considering the comparison of the distinct generalization profiles of both neural networks. Among the models, it is observed that, despite significantly higher f1-score values in the results of the CNN, the MLP, with its simpler structure, still achieves a higher level of stability as the noise increases. Nevertheless, the improvements of data augmentation with noise injection is greater for deeper neural networks than for shallow neural networks. Such results are consistent with previous work showing that both shallow and deep neural networks benefit from noise injection but improvements of deep learners are greater than those of shallow learners [12].

In the first analysis, the f1-score heatmaps showed that the CNN model (Figure 5, left) exhibits a higher Trace value in comparison to the averages of the upper and lower triangle regions. This result indicates that when the level of noise in training and testing is identical, the CNN model demonstrates the ability to learn from and leverage that specific noise profile. The situation is different for the triangle regions where the overall results were comparatively lower and the same for both upper and lower triangles.

For the MLP model (Figure 5, right), a different behaviour was observed. More stable values are seen along the direction of the lower triangle. In this case, the average values of the Trace and the lower triangle are remarkably close, while the upper triangle values are lower in comparison. These findings illustrate the adaptability of the MLP model training process in response to increasing noise levels. A shared pattern is

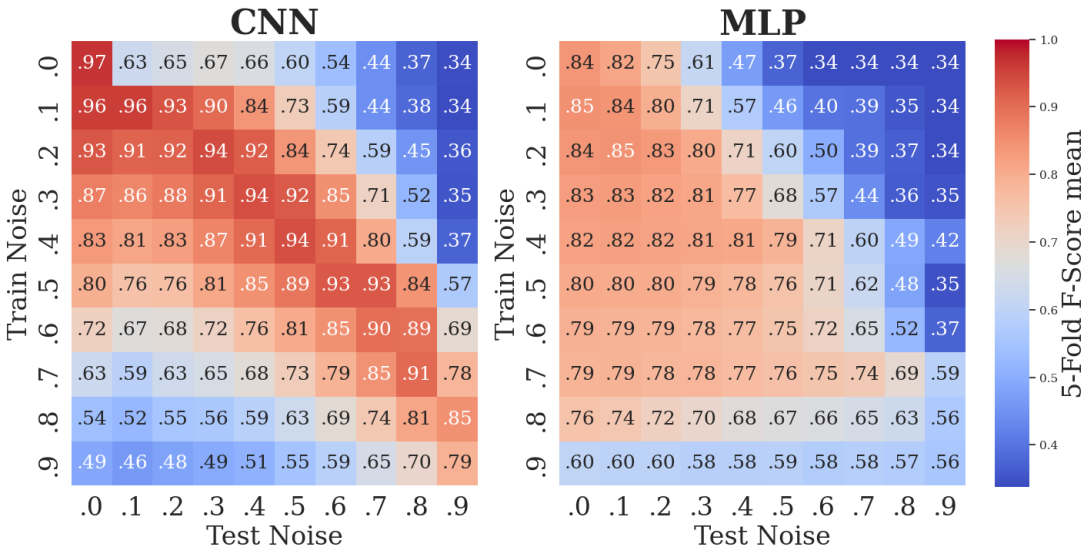

Fig. 5. F1-score results derived from 5-fold cross-validation by CNN (left) and MLP (right) models.

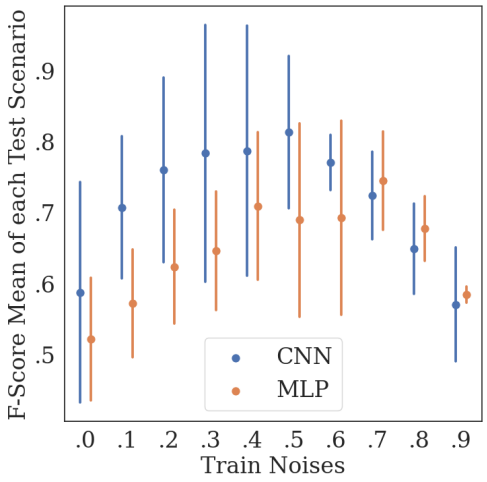

Fig. 6. Mean f1-score with 95% confidence interval for each tested scenario from both models based on 5-fold cross-validations.

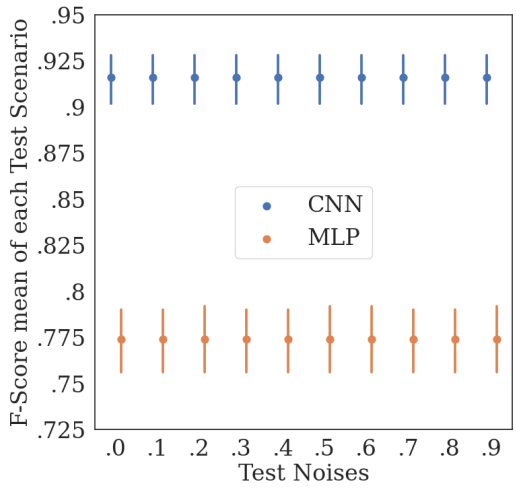

Fig. 7. F1-score results with 95% confidence interval derived from 5-fold cross validation of every test scenario, when both models were trained with 24,000 images across every noise level of this study.

evident in both cases: at the highest level of training noise investigated (90%), both the CNN and MLP models encounter significant challenges, resulting in a decline in their overall f1-score performance.

Evaluating the mean f1-score of each training scenario (Figure 6), the CNN model shows a narrowing confidence interval as the noise level increases, indicating a more consistent and robust response to escalating noise challenges. Conversely, the MLP model demonstrates a widening confidence interval, implying a greater degree of variability in its performance as the noise level escalates. This contrast underlines the distinct strategies employed by the CNN and MLP models in managing noise.

In the second analysis, where the training received every single noise level included in this study, the results suggest that the models prioritize the image information over the noise, showcasing their ability to focus on the essential features for classification tasks (Figure 7). While the overall performance may differ between the CNN and MLP, their shared consistency within each noise level emphasizes their resilience to noise and their capacity to provide reliable evaluations based on the image content.

V. Conclusion

This study contributes valuable insights into the characterization of model generalization profiles, particularly in scenarios where training and testing data distributions differ significantly due to noise levels. By implementing noise injection in MRI data for brain tumor classification, we investigated

the response of two architecturally distinct neural network models, a shallow MLP and a deeper CNN.

Our results highlight distinct differences in the generalization profiles of MLP and CNN models, each demonstrating unique strategies when confronted with noise. Despite CNN's superior performance as reflected in higher mean f1-scores, the MLP model also demonstrated resilience and adaptability as the noise levels increased. This resilience manifested in a stable f1-score even with escalating noise, indicating a robust response to noisy scenarios. It is interesting to note that under conditions of increasing disparity between training and testing distributions, simpler models like the MLP may emerge as the more suitable choice. This consideration highlights the nuanced trade-offs in model complexity and noise resilience that must be taken into account in model selection for real-world applications.

The methodology employed in this study, featuring the injection of incrementally increasing levels of noise and the utilization of f1-score heatmaps, presents a robust and versatile strategy for evaluating the generalization profile of any classification model. Notably, this methodology is adaptable and allows for modification as per the needs of specific applications. The type and level of noise injected into the dataset can be varied, providing a way to emulate different types of real-world data corruption and test the resilience of models under diverse conditions. Similarly, the performance metric used for evaluation isn't fixed to the f1-score, other metrics such as accuracy, precision, or recall could be used depending on the requirements of the application. This adaptability enhances the applicability of this approach across various domains and model architectures, offering a comprehensive way to assess model robustness and generalization capabilities in the face of noisy data.

It is essential to acknowledge several limitations of this study that warrant consideration. Firstly, the study focused solely on uniform noise, leaving out a comparison with models trained using Gaussian Noise as Noise Injection. Such comparison would have been interesting to understand the potential different behaviors of this two effects applied in the datasets. Secondly, the study primarily explored the impact of Noise Injection on a specific set of models and datasets, potentially limiting the generalizability of the findings. Including a broader range of models and datasets from different domains would have enhanced the study's applicability to various real-world applications.

REFERENCES

[1] X. Ying, "An overview of overfitting and its solutions," *Journal of Physics: Conference Series*, vol. 1168, p. 022022, feb 2019.

[2] P. Thanapol, K. Lavangnananda, P. Bouvry, F. Pinel, and F. Leprévost, "Reducing overfitting and improving generalization in training convolutional neural network (cnn) under limited sample sizes in image recognition," in *2020 - 5th International Conference on Information Technology (InCIT)*, pp. 300–305, 2020.

[3] Q. Li, M. Yan, and J. Xu, "Optimizing convolutional neural network performance by mitigating underfitting and overfitting," in *2021 IEEE/ACIS 19th International Conference on Computer and Information Science (ICIS)*, pp. 126–131, 2021.

[4] E. Ahishakiye, M. Van Gijzen, J. Tumwiine, R. Wario, and J. Obungoloch, "A survey on deep learning in medical image reconstruction," *Intelligent Medicine*, vol. 1, no. 3, 2021.

[5] S. Y. Sourab, H. R. Shuvo, R. Hasan, and T. Masruf, "Diagnosis of covid-19 from chest x-ray images using convolutional neural networking with k-fold cross validation," in *2021 IEEE International Power and Renewable Energy Conference (IPRECON)*, pp. 1–5, 2021.

[6] P. Tilekar, P. Singh, N. Aherwadi, S. Pande, and A. Khamparia, "Breast cancer detection using image processing and cnn algorithm with k-fold cross-validation," in *Proceedings of Data Analytics and Management* (D. Gupta, Z. Polkowski, A. Khanna, S. Bhattacharyya, and O. Castillo, eds.), (Singapore), pp. 481–490, Springer Singapore, 2022.

[7] M. Kusk and S. Lysdahlgaard, "The effect of gaussian noise on pneumonia detection on chest radiographs, using convolutional neural networks," *Radiography*, vol. 29, no. 1, pp. 38–43, 2023.

[8] A. Anaya-Isaza and L. Mera-Jiménez, "Data augmentation and transfer learning for brain tumor detection in magnetic resonance imaging," *IEEE Access*, vol. 10, pp. 23217–23233, 2022.

[9] R. M. Zur, Y. Jiang, L. L. Pesce, and K. Drukker, "Noise injection for training artificial neural networks: A comparison with weight decay and early stopping: Noise injection for training artificial neural networks," *Medical Physics*, vol. 36, no. 10, pp. 4810–4818, 2009.

[10] N. Levi, I. M. Bloch, M. Freytsis, and T. Volansky, "Noise injection node regularization for robust learning," 2022.

[11] X. Huang, K. Shirahama, M. T. Irshad, M. A. Nisar, A. Piet, and M. Grzegorzek, "Sleep stage classification in children using self-attention and gaussian noise data augmentation," *Sensors*, vol. 23, no. 7, 2023.

[12] Y. Bengio, F. Bastien, A. Bergeron, N. Boulanger–Lewandowski, T. Breuel, Y. Chherawala, M. Cisse, M. Côté, D. Erhan, J. Eustache, X. Glorot, X. Muller, S. Pannetier Lebeuf, R. Pascanu, S. Rifai, F. Savard, and G. Sicard, "Deep learners benefit more from out-of-distribution examples," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics* (G. Gordon, D. Dunson, and M. Dudík, eds.), vol. 15 of *Proceedings of Machine Learning Research*, (Fort Lauderdale, FL, USA), pp. 164–172, PMLR, 11–13 Apr 2011.

[13] N. Chakrabarty, "Brain mri images for brain tumor detection," 2019.

[14] J. Cheng, "Brain tumor dataset," 2017.

[15] F. Chollet *et al.*, "Keras." https://keras.io, 2015.