Revista de Informática Teórica e Aplicada - RITA - ISSN 2175-2745
Vol. 32, Num. 1 (2025) 33-39

RESEARCH ARTICLE

Preprocessing and Optimizer Impact on Image-Based Autoimmune Disease Diagnosis

Impacto do Pré-processamento e do Otimizador no Diagnóstico de Doenças Autoimunes Baseado em Imagem

João Vítor Souto de Lima[1], Rafael Marinho e Silva[2], Rodrigo Moreira[1], Larissa Ferreira Rodrigues Moreira[1]*

Abstract: The precise diagnosis of autoimmune diseases is challenging due to overlapping symptoms. Despite significant advancements in techniques based on Convolutional Neural Networks (CNNs), research opportunities remain in evaluating the impact of external factors on CNN performance. This study investigates CNN behavior by focusing on two factors: image preprocessing and optimizer choice. We assessed the effects of contrast stretching and histogram equalization on HEp-2 images and compared the Adam and SGD optimizers. Using a factorial experimental design, we trained and validated CNN models on a dataset from the International Conference on Pattern Recognition (ICPR-2014) competition, categorized into six classes of HEp-2 cells. This study contributes to understanding how different factors influence CNN models in medical image processing.
Keywords: Autoimmune Diseases — HEp-2 Cells — Classification — CNN — Factorial Design

Resumo: O diagnóstico preciso de doenças autoimunes é desafiador devido à sobreposição de sintomas. Apesar dos avanços significativos em técnicas baseadas em Redes Neurais Convolucionais (CNNs), ainda existem oportunidades de pesquisa para avaliar o impacto de fatores externos no desempenho das CNNs. Este estudo investiga o comportamento das CNNs, focando em dois fatores: o pré-processamento de imagens e a escolha do otimizador. Avaliamos os efeitos do ajuste de contraste e da equalização de histograma em imagens HEp-2 e comparamos os otimizadores Adam e SGD. Utilizando um design experimental fatorial, treinamos e validamos modelos de CNN em um conjunto de dados da competição da Conferência Internacional de Reconhecimento de Padrões (ICPR-2014), categorizados em seis classes de células HEp-2. Este estudo contribui para o entendimento de como diferentes fatores influenciam os modelos de CNN no processamento de imagens médicas.
Palavras-Chave: Doenças Autoimunes — Células HEp-2 — Classificação — CNN — Análise Fatorial

[1] *Institute of Exacts and Technological Sciences, Federal University of Viçosa (UFV), Brazil*
[2] *University Center of Patos de Minas (UNIPAM), Brazil*
*__Corresponding authors__: {rodrigo, larissa.f.rodrigues}@ufv.br
DOI: http://dx.doi.org/10.22456/2175-2745.143545 • **Received:** 24/10/2024 • **Accepted:** 02/12/2024
CC BY-NC-ND 4.0 - This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

1. Introduction

The classification of autoimmune diseases through medical imaging presents a challenge at the intersection of medicine and technology. Autoimmune diseases have seen a significant increase in recent decades, becoming the third most common cause of mortality after cardiovascular diseases and cancer. Accurate identification of these diseases is complicated by the variety of symptoms that often overlap with other conditions [1].

The use of immunofluorescence images of HEp-2 cells has become a standard procedure to assist in the diagnosis of autoimmune diseases. However, this method requires visual analyses that are subjective, highlighting the need for computer-assisted diagnostic systems that utilize image processing techniques and Artificial Intelligence (AI) [2, 3, 4].

The advancement of deep learning techniques has offered promising solutions for the automatic analysis of autoimmune diseases. However, the effectiveness of these techniques is significantly affected by the quality of image preprocessing. Inadequate preprocessing can lead to the loss of critical information, impairing diagnostic accuracy [5]. Techniques such as contrast and brightness adjustment, resizing, and filtering are essential for preparing images before training deep learning models. However, the decision to apply or omit specific preprocessing steps is not trivial and depends on the context of the application and the characteristics of the available images [6, 7].

Another challenge in applying deep learning, in particular Convolutional Neural Networks (CNNs) to medical diagnostics, is the interpretability of the models. While these models can achieve high accuracy, understanding the basis of their decisions is critical, especially in a medical context where decisions can have significant implications for patient treatment [8]. Research such as that by Barbosa et al. (2024)[9] seeks not only to optimize model performance through hyperparameter tuning but also to improve the explainability of AI models, a necessary step for their acceptance and adoption in clinical practice.

In this work, factor analysis was used to explore how different factors influence classification accuracy. Factor analysis plays a crucial role in improving the performance of deep learning models. However, to address the classification of autoimmune diseases, the scope was expanded to include additional factors, such as variations in fluorescence pattern expression and the influence of specific image preprocessing techniques tailored to autoimmune pathologies.

Furthermore, the inclusion of advanced deep learning techniques and the consideration of specific characteristics of datasets related to pathology can significantly improve the accuracy of medical image classification [10, 11]. Adapting these approaches to the domain of autoimmune diseases, combined with an analysis of the factors that impact model performance, the proposed work aims not only to advance the application of CNNs in autoimmune disease classification but also to contribute to the interpretability of Convolutional Neural Network (CNN) models using factor analysis in the field of medical image processing and automated diagnosis.

The remainder of this paper is organized as follows. Section 2 surveys the related work. Section 3 presents our proposed approach to explore the influence of preprocessing and optimizer in image based autoimmune disease. Section 4 describes our results and discusses the main findings. Finally, Section 5 presents the conclusion and opportunities for future work.

2. Related Work

In this section, recent studies applying machine learning and computer vision techniques to medical diagnosis are explored, with a focus on CNNs and image preprocessing. These works highlight the evolution of Artificial Intelligence in interpreting medical images, emphasizing the importance of preprocessing and network architectures in improving model performance.

Shayesteh et al. [12] applied texture analysis to Magnetic Resonance Imaging (MRI) to predict treatment response in patients with rectal cancer, using the Laplacian of Gaussian (LOG) filter as a preprocessing technique, which increased accuracy from 52.2% to 79.3%.

Rodrigues et al. [13][11] investigated CNNs for the analysis of immunofluorescence images, demonstrating that techniques such as contrast stretching and histogram equalization significantly improve model performance, achieving an accuracy of 98.28% with the Inception-V3 model.

Behzadi-Khormouji et al. [14] utilized CNNs for detecting consolidations in chest radiographs, surpassing architectures such as VGG16 and DenseNet121 with an accuracy of 94.67%. Sori et al. [15] developed the DFD-Net for lung cancer detection in Computerized Tomography (CT) images, increasing accuracy from 77.3% to 87.4% after preprocessing that included noise removal and image normalization.

Gautam & Raman [16] improved stroke classification accuracy using preprocessing techniques, raising accuracy from 97.5% to 98.33%. Ha et al. [17] proposed a YOLOv3-based model to detect mesiodens in panoramic radiographs, noting that the CLAHE preprocessing method resulted in lower accuracies compared to the original images, suggesting that the model performed well without significant adjustments.

Castiglione et al. [18] developed the ADECO-CNN model for COVID-19 detection in CT images, achieving an accuracy of 99.99% with image normalization. Shamila Ebenezer et al. [19] used the EfficientNet architecture for medical image classification, implementing preprocessing techniques such as gamma correction and adaptive histogram equalization, increasing accuracy from 90.15% to 94.56%.

Joseph & Oluhbara [20] investigated the impact of preprocessing on skin lesion segmentation in dermatoscopic images, using color histogram clustering with Otsu thresholding and achieving effective segmentation. Alabhadh et al. [21] utilized the Medical Vision Transformer (MVT) for skin cancer classification, observing an accuracy increase from 90.28% to 96.14% with preprocessing techniques.

Barbosa et al. [9] explored the impact of optimization factors and learning rate on CNN performance for breast cancer diagnosis, using partial factor analysis to highlight the importance of proper preprocessing and hyperparameter selection.

This work differentiates itself by analyzing the interpretability of CNNs in the classification of autoimmune disease images, using factor analysis to investigate how preprocessing affects the accuracy of CNN models, contributing to the improvement of automated diagnosis.

3. Methods

The images from the HEp-2 dataset we submitted a process of Training and Validation using five distinct CNN architectures. The CNNs were trained and evaluated according to accuracy, precision, recall, and F1-score metrics. The impact of two main factors was analyzed in the accuracy performance considering: the image preprocessing methods and the choice of optimizer during the training of the CNNs. Figure 1 summarizes our proposed method.

3.1 Dataset

The dataset used in this study was provided for the competition "Contest on Performance Evaluation on Indirect Immunofluorescence Image Analysis Systems" hosted at International

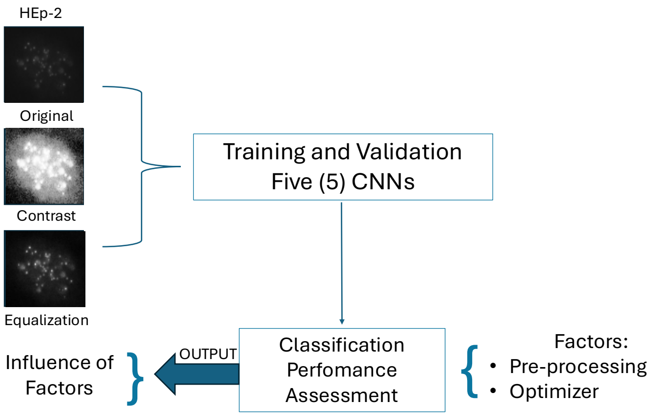
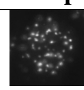
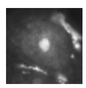
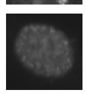
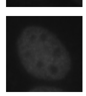


Figure 1. Assessment Method for Evaluating the Influence of factors on CNN behavior.

Conference on Pattern Recognition (ICPR-2014), and contains 13,596 images of HEp-2 cells categorized into six classes. Table 1 describes the number of images and examples of instances for each class.

Table 1. Distribution of HEp-2 cell images by class.

Class	Quantity	Example
Centromere	2741	
Golgi	724	
Homogeneous	2494	
Nucleolar	2598	
Nuclear Membrane	2208	
Speckled	2831	
Total	**13596**	

3.2 CNN Architectures

In this work, we evaluated the following CNN architectures:

- **AlexNet:** is a CNN with five convolutional layers and three fully connected layers. It uses ReLU activations and dropout to improve performance and prevent overfitting. Notably, it won the ImageNet competition by reducing error rates significantly, establishing it as a key model in image classification [6].
- **ResNet-50**: This architecture stands out for its use of residual blocks, allowing the construction of deep networks without suffering from accuracy degradation issues. ResNet-50, the winner of the ILSVRC 2015 competition, was designed to mitigate the increase in training error in deep networks, maintaining high accuracy even with a large number of layers [22].
- **VGG-16**: The VGG-16 is known for its simplicity and effectiveness, consisting of 16 layers, 13 of which are convolutional. It replaces large filters with sequences of smaller filters (3×3), which enhances the analysis of image features with a lower computational cost [23].
- **Inception V3**: An evolution of GoogLeNet, Inception V3 introduces Inception modules that perform convolution and pooling operations more efficiently, along with techniques such as dropout and ReLU activation functions to prevent overfitting. With 22 layers, this architecture offers flexibility and high efficiency in image classification [24].
- **SqueezeNet**: Focused on efficiency, SqueezeNet offers accuracy comparable to AlexNet but with significantly fewer parameters and a much smaller model size. It uses fire modules to efficiently reduce and expand feature depth, combining layers of 1×1 and 3×3 filters, and employs model compression techniques to maximize efficiency without sacrificing accuracy [25].

3.3 Preprocessing

We considered the preprocessing applied by [11] to HEp-2 images. First, contrast stretching expands the dynamic range of pixel intensities, thereby improving the visibility of distinct features in cell images. Next, histogram equalization is used to evenly redistribute the brightness across the images, increasing the accuracy of visual representation and highlighting details crucial for effective classification.

These methods were chosen with the aim of mitigating common issues in medical imaging, such as lighting variations and insufficient contrast, which can hinder the correct identification of pathological patterns.

3.4 Optimizers

In this study, we evaluated two widely used optimization algorithms: Stochastic Gradient Descent (SGD) and Adaptive Moment Estimation (Adam). SGD updates the model weights by calculating the gradient of the loss function and iteratively adjusting the weights. While effective, it can slow convergence and is sensitive to the learning rate [26].

Adam improves SGD by using adaptive learning rates for each parameter. It combines the momentum and RMSprop techniques, making it more resilient to noisy data and complex models [27]. In our evaluation, we compared the performances of both optimizers in terms of convergence speed, accuracy, and model stability.

The impact of these optimizers on accuracy, training speed, and network generalization capability was evaluated, providing insights into which optimizer performs better in the task of classifying HEp-2 cell images under the adopted preprocessing conditions.

3.5 Model Evaluation

The evaluation of the trained models was conducted on the validation and test sets using accuracy, precision, recall, and F1-score metrics. By comparing the metrics obtained on the validation and test sets, it is possible to assess the models' ability to generalize the acquired knowledge to new samples.

Accuracy, precision, recall, and F1-score are defined by Equations 1, 2, 3, e 4, respectively:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{2}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{3}$$

$$\text{F1-score} = \frac{2\times TP}{2\times TP+FP+FN} \tag{4}$$

Where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively.

3.6 Factorial Experimental Design

Factorial experimental design is a statistical technique used to identify and understand the underlying structure of a set of observed variables. The main goal of this technique is to reduce data complexity by identifying covariance patterns among the original variables and explaining these patterns through a smaller set of underlying factors [28]. The analysis helps to identify which elements have the greatest impact on the accuracy, specificity, and sensitivity of models, facilitating the optimization and selection of more effective techniques for image processing and model construction.

The factors are constructed to capture most of the total variance of the variable sets, allowing for a simpler and more meaningful interpretation of the data. The main steps involved in factor analysis include extracting factors from the correlations between variables, rotating the factors to facilitate interpretation, and identifying the factors based on their factor loadings [29].

After this analysis, a number K of factors is chosen, each with N_i levels for factor i. It is generally recommended to use a few factors with two levels per factor. For example, considering a problem with factors A and B, four experiments are conducted as shown in Table 2, yielding the values Y_1, Y_2, Y_3, and Y_4 [30].

The model for 2^2, shown in Table 2 is given by $y = q_0 + q_A \times x_A + q_B \times x_B + q_{AB} \times x_{AB}$. By substituting the four observations into the model, the values of q_0, q_A, q_B, and q_{AB} were obtained according to the following equations:

$$q_0 = \frac{1}{4}\times(y_1+y_2+y_3+y_4),$$
$$q_A = \frac{1}{4}\times(-y_1+y_2-y_3+y_4),$$
$$q_B = \frac{1}{4}\times(-y_1-y_2+y_3+y_4),$$
$$q_{AB} = \frac{1}{4}\times(y_1-y_2-y_3+y_4).$$

Table 2. Example of Factorial Partial Analysis.

Experiment	A	B	y
1	-1	-1	y_1
2	1	-1	y_2
3	-1	1	y_3
4	1	1	y_4

Thus, from the values q_0, q_A, q_B, and q_{AB}, it is possible to determine the Sum of Squares (SS) of the factors based on the Total Sum of Squares (SST = $2^2q_A^2 + 2^2q_B^2 + 2^2q_{AB}^2$) and the total variation of the response variables due to the influence and interaction between the factors [30]. From this result, it is possible to interpret the mean response variable from q_0, the variation in the response variable due to some factor, which factor the variable is more dependent on, and other insights.

The use of factorial analysis in our study enables a systematic examination of how the independent variables influence the performance of CNNs. By simultaneously considering multiple factors, such as image preprocessing techniques and optimizer selection, this method provides a comprehensive understanding of their interactions and effects on model performance. This approach is particularly important in medical applications, where small changes in image processing can significantly affect the diagnostic accuracy. Through factorial analysis, we aimed to optimize the performance of CNN and enhance our understanding of CNN behavior in medical image processing.

4. Results and Discussion

We programming the experiments using language Python and the PyTorch [31] framework due to their flexibility and comprehensive support for CNNs. All experiments were conducted on an Intel i5 3.00 GHz machine, with 48 GB of RAM and an NVIDIA GeForce RTX 4060 GPU with 8 GB of memory.

Table 3 provides a comprehensive summary of the classification performance metrics obtained from the experiments conducted in this study. The table includes accuracy, F1-score, precision, and recall values for the five CNN models evaluated using both ADAM and SGD optimizers across different image enhancement techniques (Contrast Stretching, Histogram Equalization, and Original). These results served as the basis for the subsequent analysis, where we evaluated the impact of each factor on the model classification performance.

Table 3. Summary of all experiments with Accuracy, F1-score, Precision and Recall.

		ADAM			SGD		
CNN		**Constrast Stretching**	**Histogram Equalization**	**Original**	**Contrast Stretching**	**Histogram Equalization**	**Original**
Alexnet	Accuracy (%)	89.93	22.32	21.69	96.65	95.22	95.99
	F1-score (%)	89.79	6.08	5.95	96.41	95.07	95.92
	Precision (%)	90.18	3.72	3.62	96.31	94.98	96.06
	Recall (%)	89.67	16.67	16.67	96.53	95.18	95.81
Inception_V3	Accuracy (%)	96.65	95.04	97.54	96.54	95.29	97.83
	F1-score (%)	96.38	94.64	97.45	96.27	94.95	97.45
	Precision (%)	96.66	94.36	97.64	96.44	94.79	97.15
	Recall (%)	96.13	94.97	97.28	96.16	95.14	97.79
Resnet50	Accuracy (%)	96.47	94.71	97.94	94.63	92.57	96.88
	F1-score (%)	96.18	94.69	97.57	94.07	92.39	96.59
	Precision (%)	96.25	94.67	97.92	93.98	92.77	96.51
	Recall (%)	96.12	94.71	97.26	94.20	92.12	96.68
Squeezenet1_0	Accuracy (%)	92.68	86.62	93.90	96.47	96.29	96.40
	F1-score (%)	92.09	86.53	93.43	96.35	96.05	96.18
	Precision (%)	92.75	87.79	93.47	96.39	96.37	96.82
	Recall (%)	91.65	86.48	93.45	96.33	95.78	95.66
VGG16	Accuracy (%)	86.32	20.81	19.96	97.76	96.80	98.16
	F1-score (%)	86.35	5.74	5.55	97.58	96.73	98.05
	Precision (%)	86.69	3.47	3.33	97.60	96.76	98.14
	Recall (%)	86.69	16.67	16.67	97.56	96.71	97.98

In the analysis of the experimental results, we observe distinct influences of the optimization and preprocessing factors on the accuracy of different CNN architectures. Specifically, Table 4 presents the factor levels and corresponding response variable (accuracy) for Experiment #1, while Table 5 provides the same information for Experiment #2.

Table 4 describes the factor levels for the optimizer and preprocessing techniques used in Experiment #1, with the corresponding accuracy value denoted as Y_1. The comparison of these factor levels indicates how the choice of optimizer (SGD vs. Adam) and preprocessing method (Original vs. Contrast) affects the accuracy of the CNN models.

Similarly, Table 5 outlines the factor levels for Experiment #2, where a different preprocessing method, Equalization, is introduced alongside the original Contrast method. The corresponding accuracy Y_1 is once again analyzed based on these factors, allowing for a comparison between the effects observed in the two experiments.

The results from the experiments indicate significant variations in the influence of optimization and preprocessing factors on the accuracy of different CNN architectures, as detailed in the tables of Experiments #1 and #2.

Table 4. Experiment #1 - Factors, Levels and Response Variable (Accuracy)

Factors	Levels	Accuracy
A - Optimizer	SGD (-1)	Y_1
	Adam (1)	
B - Preprocessing	Original (-1)	
	Contrast (1)	

In Table 6, SqueezeNet showed the highest dependency on the optimizer, with an influence of 92.86%. Table 7, this influence dropped to 71.14%, suggesting a lower dependency on the optimizer in this configuration. On the other hand, ResNet-50 showed an increase in optimizer influence from 36.79% in Experiment #1 to 51.9% in Experiment #2, indicating greater sensitivity to the optimizer in the second scenario.

Table 5. Experiment #2 - Factors, Levels and Response Variable (Accuracy)

Factors	Levels	Accuracy
A - Optimizer	SGD (-1)	Y_1
	Adam (1)	
B - Preprocessing	Equalization (-1)	
	Contrast (1)	

Preprocessing was highly influential for Inception-V3 in both experiments. In Experiment #1, preprocessing contributed 96.11% to accuracy, while in Experiment #2, this influence slightly increased to 98.21%. This demonstrates that, regardless of the experimental conditions, preprocessing is crucial for the performance of Inception-V3.

Notably, SqueezeNet had a very low influence from preprocessing in Experiment #1 (3.57%), but this influence increased to 15.29% in Experiment #2. This suggests that, depending on the experimental conditions, SqueezeNet can benefit from better image preprocessing.

When analyzed together, the optimizer and preprocessing showed different influences depending on the experiment. In Experiment #1, the combination had little influence on Inception-V3 (3.24%), while in Experiment #2, this influence decreased even further to 1.56%. This pattern suggests that the combination of these factors may not be as effective as the isolated application of each.

In contrast, for SqueezeNet, the combined influence of the two factors significantly increased from 3.57% in Experiment #1 to 13.58% in Experiment #2, indicating a possible synergy between the optimizer and preprocessing when applied together in this architecture.

Overall, Inception-V3 consistently remained sensitive to preprocessing in both experiments, reaffirming the importance of sophisticated preprocessing techniques for this architecture. SqueezeNet, which initially seemed more dependent on the optimizer, showed more balanced behavior in the second experiment, with significant influence from both the optimizer and preprocessing.

Table 6. Experiment #1: Influence on Accuracy (%). Factors A, B, and the Interaction of Factors A and B Simultaneously.

	Factors Influence on Accuracy		
CNN	**Optimizer (%)**	**Preprocessing (%)**	**Optimizer and Preprocessing (%)**
AlexNet	41.34	29.9	28.76
Inception-V3	0.66	96.11	3.24
ResNet-50	36.79	60.54	2.66
SqueezeNet	92.86	3.57	3.57
VGG-16	47.71	25.83	26.46

Table 7. Experiment #2: Influence on Accuracy (%). Factors A, B, and the Interaction of Factors A and B Simultaneously.

	Factors Influence on Accuracy		
CNN	**Optimizer (%)**	**Preprocessing (%)**	**Optimizer and Preprocessing (%)**
AlexNet	41.12	30.42	28.46
Inception-V3	0.24	98.21	1.56
ResNet-50	51.9	47.81	0.29
SqueezeNet	71.14	15.29	13.58
VGG-16	47.1	27.22	25.67

These results highlight the need to consider the CNN architecture when selecting preprocessing methods and optimizers. The variations observed between the experiments indicate that the effectiveness of these choices is not universal but rather dependent on the specific configuration and type of CNN used.

5. Conclusion

In this paper, we shed light on how preprocessing and optimization factors differently influence the accuracy of various Convolutional Neural Network (CNN) architectures. Inception-V3 emerged as the most sensitive to preprocessing, while SqueezeNet exhibited a mixed dependency between the optimizer and preprocessing, depending on the experimental conditions.

The findings suggest that, to improve the accuracy of neural networks in medical applications, such as autoimmune disease diagnosis, it is crucial to tailor preprocessing and optimization techniques to the specific CNN architecture in use. Additionally, understanding how these external factors impact the CNN behavior is essential for guiding future research on which aspects require further investigation, particularly in the context of real-time diagnosis. For future work, we plan to analyze additional factors, with a specific focus on neural network architectures, and evaluate different medical imaging datasets.

Acknowledgments

Rodrigo Moreira gratefully acknowledges the financial support of FAPEMIG (Grant #APQ00923-24). We also acknowledge the financial support of the FAPESP MCTIC/CGI Research project 2018/23097-3 - SFI2 - Slicing Future Internet Infrastructures. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

Author contributions

JVSL: Software, Investigation, Validation, Visualization, Writing - original draft. RMS: Investigation, Validation, Writing - Review & Editing. RM: Methodology, Investigation, Validation, Funding acquisition, Resources, Writing - Review & Editing. LFRM: Conceptualization, Methodology, Validation, Supervision, Funding acquisition, Resources, Writing - Review & Editing, Project administration.

References

[1] SMITH, J.; DOE, J. Global Rise of Autoimmune Disease. *Journal of Autoimmunity*, Elsevier, v. 102, p. 123–129, 2019.

[2] DAVIDSON, A.; DIAMOND, B. Chapter 3-general features of autoimmune disease. *The Autoimmune Diseases*, Academic Press, Boston, p. 19–37, 2014.

[3] WATANABE, K. et al. Kampo medicines for autoimmune disorders: rheumatoid arthritis and autoimmune diabetes mellitus. *Japanese Kampo Medicines for the Treatment of Common Diseases: Focus on Inflammation*, Academic Press, p. 103, 2017.

[4] ZHERNAKOVA, A.; WITHOFF, S.; WIJMENGA, C. Clinical implications of shared genetics and pathogenesis in autoimmune diseases. *Nature Reviews Endocrinology*, Nature Publishing Group UK London, v. 9, n. 11, p. 646–659, 2013.

[5] KORESH, H. J. D. et al. Impact of the preprocessing steps in deep learning-based image classifications. *National Academy of Sciences Letters*, Springer, 2024.

[6] KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, v. 25, 2012.

[7] RAZAVIAN, A. S. et al. CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. Columbus, OH, USA: IEEE, 2014. p. 512–519.

[8] ZHOU, S. K. et al. A Review of Deep Learning in Medical Imaging: Imaging Traits, Technology Trends, Case Studies With Progress Highlights, and Future Promises. *Proceedings of the IEEE*, v. 109, n. 5, p. 820–838, 2021.

[9] BARBOSA, G. et al. Optimization and Learning Rate Influence on Breast Cancer Image Classification. In: INSTICC. *Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 3: VISAPP*. Rome, Italy: SciTePress, 2024. p. 792–799. ISBN 978-989-758-679-8.

[10] GAO, Z. et al. HEp-2 cell image classification with deep convolutional neural networks. *IEEE journal of biomedical and health informatics*, IEEE, v. 21, n. 2, p. 416–428, 2016.

[11] RODRIGUES, L. F.; NALDI, M. C.; MARI, J. F. Comparing convolutional neural networks and preprocessing techniques for HEp-2 cell classification in immunofluorescence images. *Computers in Biology and Medicine*, v. 116, p. 103542, 2020. ISSN 0010-4825. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S0010482519303993⟩.

[12] SHAYESTEH, S. P. et al. Prediction of Response to Neoadjuvant Chemoradiotherapy by MRI-Based Machine Learning Texture Analysis in Rectal Cancer Patients. *Journal of Gastrointestinal Cancer*, v. 51, n. 2, p. 601–609, 2020. ISSN 1941-6636. Disponível em: ⟨https://doi.org/10.1007/s12029-019-00291-0⟩.

[13] RODRIGUES, L. F.; NALDI, M. C.; MARI, J. F. HEp-2 Cell Image Classification Based on Convolutional Neural Networks. In: *2017 Workshop of Computer Vision (WVC)*. [S.l.: s.n.], 2017. p. 13–18.

[14] BEHZADI-KHORMOUJI, H. et al. Deep learning, reusable and problem-based architectures for detection of consolidation on chest X-ray images. *Computer Methods and Programs in Biomedicine*, v. 185, p. 105162, 2020. ISSN 0169-2607. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S0169260719306960⟩.

[15] SORI, W. J. et al. DFD-Net: lung cancer detection from denoised CT scan image using deep learning. *Frontiers of Computer Science*, v. 15, n. 2, p. 152701, 2020. ISSN 2095-2236. Disponível em: ⟨https://doi.org/10.1007/s11704-020-9050-z⟩.

[16] GAUTAM, A.; RAMAN, B. Towards effective classification of brain hemorrhagic and ischemic stroke using cnn. *Biomedical Signal Processing and Control*, v. 63, p. 102178, 2021. ISSN 1746-8094. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S1746809420303153⟩.

[17] HA, E.-G. et al. Automatic detection of mesiodens on panoramic radiographs using artificial intelligence. *Scientific Reports*, v. 11, n. 1, p. 23061, 2021. ISSN 2045-2322. Disponível em: ⟨https://doi.org/10.1038/s41598-021-02571-x⟩.

[18] CASTIGLIONE, A. et al. COVID-19: Automatic Detection of the Novel Coronavirus Disease From CT Images Using an Optimized Convolutional Neural Network. *IEEE Transactions on Industrial Informatics*, v. 17, n. 9, p. 6480–6488, 2021.

[19] Shamila Ebenezer, A. et al. Effect of image transformation on efficientnet model for covid-19 ct image classification. *Materials Today: Proceedings*, v. 51, p. 2512–2519, 2022. ISSN 2214-7853. International Conference on Advances in Materials Science. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S2214785321078469⟩.

[20] JOSEPH, S.; OLUGBARA, O. O. Preprocessing effects on performance of skin lesion saliency segmentation. *Diagnostics*, v. 12, n. 2, 2022. ISSN 2075-4418. Disponível em: ⟨https://www.mdpi.com/2075-4418/12/2/344⟩.

[21] ALADHADH, S. et al. An effective skin cancer classification mechanism via medical vision transformer. *Sensors*, v. 22, n. 11, 2022. ISSN 1424-8220. Disponível em: ⟨https://www.mdpi.com/1424-8220/22/11/4008⟩.

[22] HE, K. et al. Deep Residual Learning for Image Recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, 2016. p. 770–778.

[23] SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[24] SZEGEDY, C. et al. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, v. 31, n. 1, Feb. 2017. Disponível em: ⟨https://ojs.aaai.org/index.php/AAAI/article/view/11231⟩.

[25] IANDOLA, F. N. et al. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and ¡0.5mb model size. *arXiv:1602.07360*, 2016. Disponível em: ⟨https://arxiv.org/abs/1602.07360⟩.

[26] LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, IEEE, v. 86, n. 11, p. 2278–2324, 1998.

[27] KINGMA, D. P.; BA, J. *Adam: A Method for Stochastic Optimization*. 2017. Disponível em: ⟨https://arxiv.org/abs/1412.6980⟩.

[28] LAWLEY, D. N.; MAXWELL, A. E. Factor analysis as a statistical method. *Journal of the Royal Statistical Society. Series D (The Statistician)*, [Royal Statistical Society, Wiley], v. 12, n. 3, p. 209–229, 1962. ISSN 00390526, 14679884. Disponível em: ⟨http://www.jstor.org/stable/2986915⟩.

[29] HAIR, J. F. et al. *Multivariate data analysis (4th ed.): with readings*. USA: Prentice-Hall, Inc., 1995. ISBN 0023490209.

[30] MOREIRA, R. et al. Deploying scalable and stable xdp-based network slices through nasor framework for low-latency applications. In: BAROLLI, L.; WOUNGANG, I.; ENOKIDO, T. (Ed.). *Advanced Information Networking and Applications*. Cham: Springer International Publishing, 2021. p. 715–726. ISBN 978-3-030-75075-6.

[31] PASZKE, A. et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019. ⟨https://pytorch.org/docs/stable/index.html⟩. Accessed: 2024-04-18.