"LeukNet" - A Model of Convolutional Neural Network for the Diagnosis of Leukemia

Luis H. S. Vogado[1]
Departamento de Computação
Universidade Federal do Piauí
Teresina, Brasil
lhvogado@gmail.com

Rodrigo de M. S. Veras (Orientador)
Departamento de Computação
Universidade Federal do Piauí
Teresina, Brasil
rveras@ufpi.edu.br

Kelson R. T. Aires (Coorientador)
Departamento de Computação
Universidade Federal do Piauí
Teresina, Brasil
kelson@ufpi.edu.br

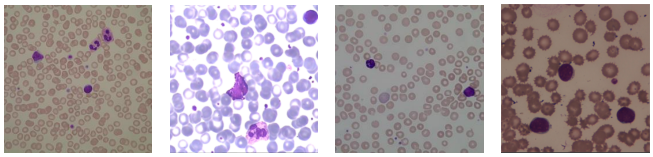
***Abstract*—Leukemia is a disorder that affects the bone marrow, causing uncontrolled production of leukocytes, impairing the transport of oxygen and causing blood coagulation problems. In this article, we propose a new computational tool, named LeukNet, a Convolutional Neural Network (CNN) architecture based on the VGG-16 convolutional blocks, to facilitate the leukemia diagnosis from blood smear images. We evaluated different architectures and fine-tuning methods using 18 datasets containing 3536 images with distinct characteristics of color, texture, contrast, and resolution. Additionally, data augmentation operations were applied to increase the training set by up to 20 times. The k-fold cross-validation (k = 5) results achieved 98.28% of accuracy. A cross-dataset validation technique, named Leave-One-Dataset-Out Cross-Validation (LODOCV), is also proposed to evaluate the developed model's generalization capability. The accuracy of using LODOCV on the ALL-IDB 1, ALL-IDB 2, and UFG datasets was 97.04%, 82.46%, and 70.24%, respectively, overcoming the current state-of-the-art results and offering new guidelines for image-based computer-aided diagnosis (CAD) systems in this area.**

I. Introduction

Leukemia is one of the most dangerous diseases according to the American Cancer Society, with an estimate of 61,780 new cases and 22,840 deaths in 2019. This disease has unknown cause and affects the production of white blood cells in the bone marrow. Due to the disease, young cells or blasts are produced abnormally, replacing healthy blood cells, i.e., white blood cells, red blood cells and platelets. Consequently, the affected individual suffers from oxygen transport problems and infections. Among the forms of diagnosis of leukemia, one can find the lumbar puncture, myelogram, blood count and flow cytometry. Samples of blood smears with healthy and unhealthy leukocytes are shown in Figure 1.

Computer aided diagnosis (CAD) systems aim to assisting medical specialists by offering information that help on their diagnosis [2]. These systems could be employed to the screening of diseases, providing a first diagnosis, or to offer a second opinion based on previously labelled examples.

One of the main issues in recent studies addressing medical imaging applications is the lack of heterogeneity in the image datasets that are used to evaluate the methods [3]. The image datasets are often acquired using similar equipment, sampled from particular populations and annotated by a limited group of specialists. Evaluation based on hold-out or cross-validation may be adequate to validate the performance within a dataset, but it is unclear how it generalizes for other datasets. Considering Deep Learning methods, this is even more relevant, since it is known that models with sufficiently large capacity may be able to specialize to the training data and fail to generalize. Although transfer learning methods were shown to be useful in many applications, there is a relevant interest on studying how to choose the proper architectures and training strategies that remain the built models useful within the same domain of application but with changes, for example, as to the source of images, sensors, viewpoint and acquisition setup [4]. Therefore, there is a gap in the literature about guidelines for the design and evaluation of CAD systems that are consistent, robust and reliable to be used in practice.



(a) (b) (c) (d)

Fig. 1. Examples of images used in this work: (a) [1] and (b) [1] healthy examples, (c) [1] and (d) [1] unhealthy examples.

In this context, we propose LeukNet, which is based on a Convolutional Neural Network (CNN), but that uses transfer learning concepts selected according to an extensive study of architectures that was made and advanced training strategies, and a in-depth discussion of evaluation. A modified Deeply Fine-Tuning (mDFT) method is employed in the training of the proposed model. The experimental results shown the need for an evaluation protocol using Leave-One-Dataset-Out Cross Validation (LODOCV), where the test is carried in one dataset, while the remaining datasets are used in the training process. This procedure is performed until all datasets are tested individually. This ensures that the CNN is not trained with any image of the datasets to be tested.

The remainder of the article is organized as follows: a description of related work is given in Section II along with

[1]M.Sc. dissertation.

the contributions achieved with this work. In Section III, the material and methods used are described, including the proposed LeukNet. In Section IV, the results as well as their discussion are presented. Finally, conclusions and perspectives for future work are given in Section V.

II. Related work and contributions

Related work is discussed in this section, in particular with respect to the image descriptor employed, the sample size, validation method and accuracy (A) results. Table I lists the works identified and their main characteristics. We identified two mains approaches: handcrafted features, deep learning models.

From Table I, one can note that only Vogado et al. [5] used more than two datasets in the experiments. Still, in Table I, it is possible to verify that the used evaluation protocols are usually the holdout and k-fold within the same dataset. When using CNN's, holdout was the most used technique. Due to availability of relatively small image datasets, one might question the convergence of the used classifiers and their ability to generalize since the relationship between number of instances used for training and the complexity of the built model falls short in such scenarios. In this context, using more datasets would allow to better evaluate the systems and their robustness in terms of considering different sources of images.

TABLE I
SUMMARY OF RELATED WORK (A - ACCURACY).

Work	Year	Descriptor	Classifier	Images	Validation	A(%)
Patel e Mishra [6]	2015	Shape, texture, statistical and color features	SVM	27	holdout	93.75
Singhal et al. [7]	2016	Texture features	SVM	260	k-fold	93.80
Thanh et al. [8]	2018	Deep features	CNN	1188	holdout	96.60
Shafique et al. [9]	2018	Deep features	CNN	760	holdout	99.50
Rehman et al. [10]	2018	Deep features	CNN	330	holdout	97.78
Sipes et al. [11]	2018	Deep Features	CNN	388	holdout	88.00
Vogado et al. [5]	2018	Deep Features	SVM	1268	k-fold	99.76
Pansombut et al. [12]	2019	Deep Features	CNN	363	holdout	81.74
Ahmed et al. [13]	2019	Deep Features	CNN	903	k-fold	88.25

III. Materials and Methods

An extensive study of architectures and training strategies was performed in order to design the network f to be used. As a result, transfer learning from five pre-trained architectures and four fine-tuning techniques were employed. The impact of data augmentation in the classification problem under study was also investigated.

A. Image Datasets

To evaluate the generalization capability of the proposed system, we used 18 public datasets, divided into development and performance sets.

Through experimentation, we used the development set to define the ideal configuration of the proposed model. This set was formed by 17 datasets, totalling 3415 images. Those images present heterogeneity in terms of color, contrast, resolution and texture, and each dataset has different balance ratio between classes, which put into test the robustness of the proposed classifier.

The performance set is a novel dataset, acquired at the Federal University of Goiás (UFG), in Brazil, referred here as UFG dataset[1]. This dataset has 121 images acquired using different microscopes, and with distinct characteristics of color, texture and contrast. This is the first article reporting results using this image dataset.

From the datasets used in our experiments, only three are class-balanced: UFG, ALL-IDB1 and ALL-IDB2 [14], as reported in Table II. Some of them have only one leukocyte per image and others have multiple leukocytes per image. Only UFG and Bloodline datasets have these two patterns.

TABLE II
SUMMARY OF THE IMAGE DATASETS USED IN THE EXPERIMENTS.

Dataset	Non-pathological	Pathological	Total	Ref.
ALL-IDB 1	59	49	108	[14]
ALL-IDB 1 (Crop)	0	510	510	[14]
ALL-IDB 2	130	130	260	[14]
Leukocytes	149	0	149	[15]
CellaVision	109	0	109	[16]
Atlas	0	88	88	-
Omid et al. 2014	154	0	154	[17]
Omid et al. 2015	0	27	27	[18]
ASH	0	96	96	[1]
Bloodline	0	204	204	[19]
ONKODIN	0	78	78	[20]
CellaVision 2	100	0	100	[21]
JTSC	300	0	300	[21]
UFG	57	64	121	-
PN-ALL Dataset	0	30	30	[22]
leukemia-images	0	140	140	link
MIDB Dataset	0	673	673	link
LISC Dataset	376	0	376	[23]
Total	**1434**	**2102**	**3536**	-

B. Data Augmentation

Improving the generalization of the deep learning models is one of the challenges in this area, but Data Augmentation is a powerful way to overcome this [24]. Augmented data is expected to represent a more extensive data set, minimizing the differences between the training and validation sets as well as any future test sets [25].

In this work, the image development set is relatively balanced. It contains 1001 non-pathological and 1182 pathological images. Therefore, data augmentation was applied equally in both classes.

The augmentation operations used were: rotation in the range of 0 to 40º, vertical, horizontal, shear and zoom in the range of 0 to 0.2, as well as horizontal and vertical flip. Notice that the nuclei images do not have asymmetry allowing flipping in both directions. Hence, the reflection fill operation was applied to replace black pixels resulting from rotation and translation techniques. Finally, the pixels of input images were normalized to values between 0 (zero) and 1 (one). The augmentation resulted in a dataset 20 times larger than the original dataset.

C. Transfer Learning

Transfer learning techniques often employed for convolutional networks uses weights that are pre-trained in large datasets, such as the ImageNet Challenge dataset [26]. This procedure decreases the requirement to retrain all parameters

[1]https://hematologia.farmacia.ufg.br (accessed in June 2020)

of a CNN from scratch [27]. Figure 2 depicts this idea. Note that some layers are copied from the pre-trained network, forming a base architecture, while other layers are randomly initialized and customized to the task at hand.

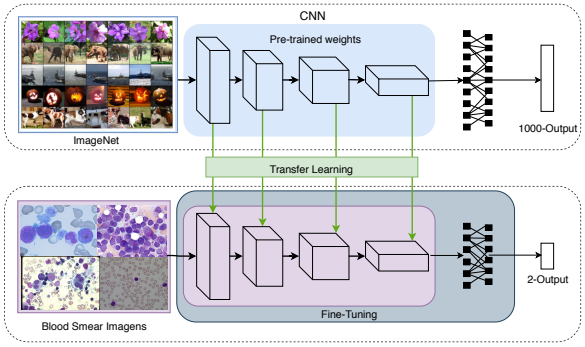


Fig. 2. Transfer learning and fine-tuning used in the development of the proposed CNN model.

Two approaches are often employed when using pre-trained weights: One approach is to extract features as the activation maps of the pre-trained network layers, defining those as feature vectors to be used as input to shallow classifiers, such as SVM [5]. The other one is to perform fine-tuning by creating a new classification layer.

According to Tajbakhsh et al. [28] and Izadyyazdanabadi et al. [29], there are two types of fine-tuning, Shallow Fine-Tuning (SFT) and Deeply Fine-Tuning (DFT). SFT consists of freezing layers from the beginning of CNN, usually the first convolutional layers, that are considered more general and allow representing shape, texture and color. The top layers are often domain-specific, carrying semantic content from the instance labels. Therefore, SFT fine-tuning provides greater specialization in the later layers, while keeping the first ones.

The DFT approach allows training the entire network, adapting even the first layers. Although it requires higher computational cost and a larger amount of data, it can benefit applications where the target domain differs from the one used to pre-train the weights.

Previous studies report better results in small datasets with smaller network architectures, in particular for binary classification [30]. Therefore, the experiments performed, alternatives to the SFT and DFT approaches, referred to as modified Shallow (mSFT) and Deeply Fine-Tuning (mDFT), respectively, were developed. In those approaches, we replaced dense layers – prior to the output layer – with new ones with smaller dimensionality (we evaluated layers with 256, 512 and 1024 elements). This decreases the number of parameters of the network, allowing faster training and making it less prone to overfitting.

D. Evaluated Architectures

The CNN architectures designed for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [26] were explored. Sequential networks such as VGG-16 and VGG-19 facilitate changes in the architecture structure while Residual and Inception-based networks presented better results in the ILSVRC. ResNet50 and InceptionV3 have less parameters than VGGNets, but have more depth, as can be verified in Table III. The architectures are indicated in Table III in terms of: year of publication, topological depth of the network (including batch normalization, activation layers, etc.), accuracy and errors in ImageNet.

TABLE III
CHARACTERISTICS OF THE EVALUATED DEEP LEARNING MODELS.

Model	Topological Depth	Number of parameters	Year
VGG-16 [31]	23	138,357,544	2014
VGG-19 [31]	26	143,667,240	2014
ResNet50 [32]	168	25,636,712	2015
InceptionV3 [33]	159	23,851,784	2016
Xception [34]	126	22,910,480	2017

In mSFT and mDFT approaches, the initial size of fully connected layers was based on Zang et al. [35] that studied cervical cancer images, which are similar to those for leukemia diagnosis, and employed layers with dimensionality 1024 and 256.

To evaluate the Inception architectures, we used the mSFT and mDFT approaches because when we fine-tuned InceptionV3, we added a new layer of Global Average Pooling, as well as a dense layer with 1024 elements with ReLu activation. For Xception, we added a dense layer with 128 elements. The output layer is the same as that presented in sequential architectures.

In the ResNet model, we performed the same process as in the InceptionV3. In the mSFT, only the added layers were trained, freezing the previous ones. On the other hand, in mDFT, all parameters were allowed to be fine-tuned.

E. Validation Methodology (Leave-one-dataset-out)

According to Diaz-Pinto et al. [36], CNNs take into account only the raw pixel information to classify images. It is expected that the accuracy will be significantly affected when the model receives as input an image from a different dataset from those used in CNN training or validation. Given this, methods that classify well images of a dataset will not necessarily succeed in images of other datasets. Thus, a critical experiment to evaluate the classifier performance is to use images obtained from different datasets.

We consider that k-fold cross validation within a given dataset does not simulate the conditions of a real scenario. When applying folding division, similar examples from the same dataset are likely to be present in both training and test sets.

Thus, we employed the Leave one dataset out cross validation (LODOCV), a validation for systems that operate on several datasets from different sources. Considering that the number of datasets available is d, we use $d - 1$ datasets for training, and evaluate the method in the unseen dataset. We repeated this experiment until all datasets were tested individually. This strategy ensured that none of the images in a dataset was present in both training and testing.

From Table II, it is possible to confirm that 15 image datasets present only a single class. Thus, datasets with images of both classes were used in the test step in LODOCV. Therefore, we performed three main experiments with ALL-IDB 1, ALL-IDB 2 and UFG datasets. For example, in the first experiment, we used ALL-IDB 1 as a test set, while the other 17 datasets were used in the training. This process was repeated for ALL-IDB 2 and UFG datasets.

IV. Experiments

We performed experiments and evaluated the results in terms of accuracy (A), precision (P), recall (R), specificity (S) and loss value. Because the new layers are trained from randomly initialized weighs, we performed five runs to compute mean and standard deviation of the metrics. All experiments were carried out on a PC with a 3.6 GHz Intel®Xeon™processor, 24 GB RAM, and a NVIDIA TITAN XP 12GB graphics card.

A. Models and Fine-Tuning Evaluation

We used an ablation study to define the base architecture of the proposed model and the training methodology. Through validation by LODOCV, we evaluated the development set through experiments with the ALL-IDB 1 and ALL-IDB 2 datasets.

Table IV indicates the best results obtained in architectures and fine-tuning techniques evaluated on the bases ALL-IDB 1 and ALL-IDB 2.

TABLE IV
BEST RESULTS OBTAINED IN ARCHITECTURES AND FINE-TUNING TECHNIQUES EVALUATED ON THE BASES ALL-IDB 1 AND ALL-IDB 2.

Arq.	Fine-tuning	A(%)	P(%)	R(%)	S(%)
ALL-IDB 1					
VGG-16	mDFT	97.04±1.21	96.42±2.45	97.14±1.83	96.95±2.21
VGG-19	DFT	97.04±0.41	95.98±0.04	97.55±0.91	96.61±0.00
Inception V3	mDFT	65.56±9.79	58.47±17.18	73.92±16.99	60.91±19.90
Xception	mDFT	77.41±8.65	69.40±9.76	94.69±3.10	63.05±18.11
ResNet50	mDFT	87.96±2.70	91.59±8.56	82.04±4.42	92.88±8.08
ALL-IDB 2					
VGG-16	mDFT	82.46±0.02	77.59±0.04	92.30±0.08	72.61±0.09
VGG-19	mDFT	79.62±6.31	77.54±8.53	85.38±9.53	73.85±12.93
InceptionV3	mDFT	58.38±3.09	56.95±2.42	70.00±12.38	46.77±11.66
Xception	mDFT	64.92±2.60	63.58±2.54	70.31±7.06	59.54±6.45
ResNet50	mDFT	69.46±6.26	66.96±7.05	80.31±10.17	58.62±16.71

From the data shown, one can verify that mDFT achieved high rates compared with the other approaches. When comparing the accuracy obtained by InceptionV3, Xception and ResNet50 architectures, it is possible to verify that ResNet50 achieved better results in both datasets. However, when compared those outcomes with the ones obtained using sequential architectures (VGG 16 and VGG 19), one can see a decrease in performance. Therefore, it is possible to conclude that this is because these architectures deal better with greater complexity in terms of the amount of data and classes than the other ones.

LeukNet was designed after analyzing the previously described results, where VGG-16 and VGG-19 architectures presented the best outcomes, with similar values for the mDFT approach in the ALL-IDB2 dataset. Therefore, we performed the Student's t-test [37] to statistically compare the results at a significance level of 5%. From the test performed, we found that the results were equivalent. Therefore, we selected VGG-16 due to its smaller number of trainable parameters.

According to Kornblith et al. [38], the best performing architectures on ImageNet can provide better feature extraction and fine-tuning. However, the authors observed this fact only in photographic databases. In datasets with fine-grained images, the effects of pre-training with ImageNet were considered small. This study indicated that the features obtained from ImageNet are not adequately transferred to such datasets. According to Sipes et al. [11] leukemia images are considered fine-grained images. This fact explains why the results achieved by VGG-16 and VGG-19 were superior to the other CNNs.

We also performed experiments varying the size of the fully connected layers to find the best compromise between accuracy and loss (see Table V), which allowed to find the highest accuracy with 1024 and 256 neurons.

TABLE V
RESULTS OBTAINED USING DIFFERENT DIMENSIONALITIES FOR LEUKNET'S FULLY CONNECTED LAYERS.

Fc Layers	A(%)	P(%)	R(%)	S(%)
ALL-IDB 1				
512-256	94.81±2.41	93.92±2.50	94.69±3.09	94.91±2.07
1024-256	97.04±1.21	96.42±2.45	97.14±1.83	96.95±2.21
1024-512	93.14±3.73	92.95±7.62	92.65±3.09	93.55±7.89
1024-1024	93.70±1.65	91.79±3.11	94.69±1.82	92.88±3.03
ALL-IDB 2				
512-256	71.53±4.97	70.91±4.20	74±13.98	69.07±9.90
1024-256	82.46±0.02	77.59±0.04	92.30±0.08	72.61±0.09
1024-512	71.84±3.64	77.84±12.16	67.53±17.09	76.15±23.46
1024-1024	69.15±2.11	69.91±4.94	69.23±9.41	69.07±11.52

We used the Student's t-test to identify if the accuracy achieved by the layers 1024-256 and 1024-512 are similar. According to the test, the results are equivalent considering the true null hypothesis. Therefore, we selected layers with size 1024-254 because they presented the best compromise between the number of parameters and accuracy.

Figure 3 depicts, as heat maps, the output of some LeukNet's convolutional filters. It can be seen that CNN excludes the background and defines the cytoplasm and leukocyte nucleus as regions of interest. However, the nuclei region (regions in yellow tone in Figure 3) is considered here as the most crucial region for classification.

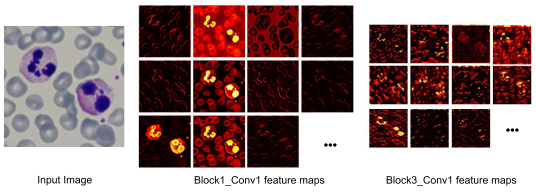


Fig. 3. Heatmap of some LeukNet's convolutional filters output.

B. Proposed model: LeukNet

The best built model has five convolutional blocks and two fully connected layers. After each convolutional block, max pooling is employed. The first two blocks have only two

convolutional layers while the remaining have three layers. The first block has 64 filters with size 3×3. From the second block on, the amount of filters is doubled, to 128, and after the convolution, the pooling operation reduces the filter size. Finally, the last two convolutional blocks have the same amount of filters. Figure 4 shows the final structure of the proposed model.

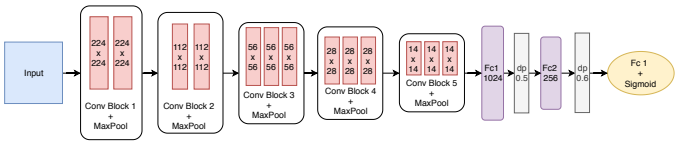


Fig. 4. Detailed structure of the proposed CNN after performing the fine-tuning.

To avoid overfitting, dropout (dp) was also employed after each fully connected layer with rates of 0.5 and 0.6, respectively. Since we deal with a binary classification problem, the output layer has one neuron with sigmoid activation function.

The Stochastic Gradient Descent (SGD) optimization algorithm was employed with batch size 32 and for a total of 50 epochs. Therefore, we used 0.001 and 0.8 for the learning rate and the momentum, respectively. The loss function used during fine-tuning was the binary cross-entropy to allow computing the gradients at each iteration.

C. Beyond CNN results with a Features Space Analysis

In order to go beyond the results obtained by fine-tuning the CNNs, we carried out two additional analyses using the features spaces formed by two models. In particular, the goal was to compare the models in terms of the linear separability of the feature spaces generated by the layer prior to the network classifier (output layer). Because we employed a linear SVM classifier, which has strong learning guarantees, better results would favor models with better generalization capabilities [39].

The analyses were performed for two scenarios. The former consists of validation with LODOCV using feature extraction with pre-trained VGG-16 on ImageNet and fine-tuned VGG-16. The second scenario uses the LODOCV test as input to the k-fold cross-validation with $k = 10$. Both experiments used the same pre-trained models for feature extraction.

For the model pre-trained with ImageNet and those refined with DFT technique, the output vector had 4096 features. Therefore, to analyze the intrinsic dimensionality in the data, we applied Principal Component Analysis (PCA) and reduced the vector to its 100 principal components. Table VI presents the results obtained by the two performed analyses.

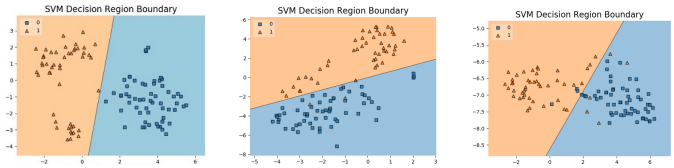
From the results in Table VI, it is possible to realize that in experiments with multiple datasets (LODOCV experiment), mDFT provided a superior linear separability of the data. However, for only one dataset (k-fold experiment), the DFT showed better results. The advantage of mDFT in the first experiment was because it restricts dense layers, in terms of dimensionality (from 4096 in the original model to 256 in the proposed model), making the model robust to images from different datasets. The DFT uses a larger output, it consequently has more "degrees of freedom" in the pre-trained model, which can cause overfitting in datasets used for fine-tuning, reducing the accuracy in an experiment with multiple datasets.

TABLE VI
FEATURE SPACE ANALYSIS PERFORMED FOR THE VGG-16 ARCHITECTURE.

		LODOCV				*k-fold*			
Approach	**Num. of Features**	**A (%)**	**P (%)**	**R (%)**	**Kappa**	**A (%)**	**P (%)**	**R (%)**	**Kappa**
ALL-IDB 1									
DFT	100	59.25%	52.68%	100%	0.2362	99.07%	98%	100%	0.9813
mDFT	256	87.96%	86.00%	87.75%	0.7575	91.66%	95.45%	83.04%	0.8571
ImageNet	100	68.51%	59.49%	95.91%	0.3962	97.22%	96%	97.95%	0.9440
ALL-IDB 2									
DFT	100	51.92%	51.06%	92.30%	0.0384	94.23%	93.89%	94.61%	0.8846
mDFT	256	73.84%	75.83%	70.00%	0.4769	85.00%	84.21%	86.15%	0.7
ImageNet	100	49.61%	49.68%	60%	-0.007	87.69%	87.69%	87.69%	0.7538

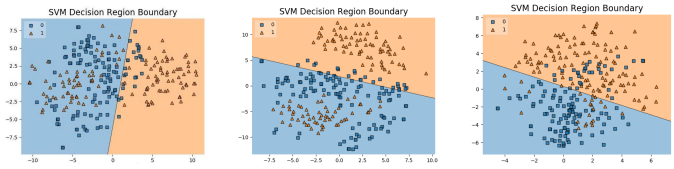
This analysis confirms previous findings that indicate that models with a more restricted bias, i.e., in terms of their space of admissible functions, may transfer better for different domains [4] in comparison to the same domain, which in the case of the widely used ImageNet dataset are mostly natural images and photographic data.

In addition to the classification experiments, we also visualized the feature spaces using a t-SNE projection, with the respective decision boundaries estimated to the 2D case, both for ALL-IDB 1 (see Figure 5) and ALL-IDB 2 (see Figure 6). From these figures, it is possible to note how the decision boundaries show good discrimination capability of the feature spaces. Also, it is clear how ALL-IDB2 is a more challenging dataset, and that the mDFT tend to produce a space that better separate the classes in comparison to the greater class overlap shown in DFT and no-finetuning spaces (see Figure 6).



(a) no fine-tuning (b) DFT (c) mDFT

Fig. 5. ALL-IDB 1 dataset visualizations using t-SNE projection in 2D along with the estimated decision boundaries using Linear SVM classifiers for different feature extraction methods: (a) no fine-tuning, (b) DFT and (c) mDFT.



(a) no fine-tuning (b) DFT (c) mDFT

Fig. 6. ALL-IDB 2 dataset visualizations using t-SNE projection in 2D along with the estimated decision boundaries using Linear SVM classifiers for different feature extraction methods: (a) no fine-tuning, (b) DFT and (c) mDFT.

D. Discussion

The results presented in Section IV-A were obtained using the LODOCV. However, other literature works do not use this validation. Thus, we applied k-fold cross-validation, with $k = 5$, to compare the results of the proposed approach with the ones obtained by sate of art methods. Table VII presents the results achieved by the proposed approach; the indicated accuracy values for the other methods were taken from their original articles.

TABLE VII
COMPARISON BETWEEN THE RESULTS OBTAINED BY THE PROPOSED METHOD AND STATE OF THE ART METHODS.

Work	**Images**	**Validation technique**	**A**(%)
Handcrafted features			
Patel e Mishra [6]	27	holdout	93.75
Singhal et al. [7]	260	k-fold	93.80
Deep-Learning-based systems			
Thanh et al. [8]	1188	holdout	96.60
Shafique et al. [9]	760	holdout	99.50
Rehman et al. [10]	330	holdout	97.78
Sipes et al. [11]	388	holdout	88.00
Pansombut et al. [12]	363	holdout	81.74
Feature extraction with CNNs			
Vogado et al. [5]	1268	k-fold	**99.76**
Ahmed et al. [13]	903	k-fold	88.25
LeukNet	3536	k-fold	**98.24**

First, from Table VII, it is possible to verify that the amount of images used in all competing methods is inferior to those presented in our experiments.

Among the studies studied, the method of Vogado et al. [5] presented experiments in more than two databases: eight of the fourteen that were used in this work. Given that this method was the only showing performance higher than the one of the proposed method, we performed a more detailed comparison between the two methods. Table VIII presents the comparative result of the proposed method and the one suggested by Vogado et al. [5] using k-fold cross-validation in the 3536 images of the 18 datasets available.

TABLE VIII
COMPARISON BETWEEN THE PROPOSED METHOD (LEUKNET) AND THE METHOD SUGESTED BY VOGADO ET AL. [5] WITH K-FOLD CROSS-VALIDATION.

Approach	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)
Vogado et al. [5]	92.79	92.90	92.80	92.22
LeukNet	**98.24**	**98.20**	**98.76**	**97.34**

Vogado et al. [5] used eight of the fourteen datasets used in this study. Comparing the results presented in Tables VII and VIII, one can notice that there was a reduction in accuracy (from 99.76 to 92.79%) due to the inclusion of new images. It is possible to highlight the ASH, UFG, Bloodline and ONKODIN datasets, which were created with images from several microscopes, with distinct resolutions, textures and different color characteristics.

To expand the comparison, Table IX shows the comparative result of the proposed method and the one suggested by Vogado et al. using the UFG performance set. One can observe that the two approaches achieved lower results when evaluated by k-fold cross-validation. However, the performance decrease of the proposed method was more moderate (from 98.28 to 70.24%) than that of the method developed by Vogado et al. [5] (from 92.79 to 52.06%). This result demonstrates that LeukNet generalizes better than the competing methods. One can believe that this result is due to the use of data augmentation techniques and a precise definition of the network parameters.

TABLE IX
COMPARING THE PROPOSED MODEL WITH THE METHOD SUGGESTED BY VOGADO ET AL. [5] BY LODOCV IN THE UFG DATASET.

Approach	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)
Vogado et al. [5]	52.06	49.90	52.10	47.70
LeukNet	**70.24**$\pm$5.51	**70.54**$\pm$8.62	**80.31**$\pm$15.17	**58.94**$\pm$22.24

V. CONCLUSION

In this work, a novel CNN architecture and training strategy were presented for the diagnosis of leukemia in blood smear images. Different architectures, parameters and fine-tuning scheme were studied to define our model. This allowed to develop a model for diagnosis that is more precise and robust than the state of the art works.

From the comparison performed against previous studies, some conclusions may be drawn as to leukemia diagnosis from images: First, fine-tuning may be more efficient than off-the-shelf feature extraction. Second, CNNs with more representations through feature maps prove better in cross-dataset experiments. Also, the choice of fine-tuning technique is essential for the correct definition of CNN parameters. Since blood sample images belong to a different domain to those used to pre-train the layers, adjusting all layers is preferable.

The use of the LODOCV evaluation demonstrated the need for more challenging experiments towards a better generalization capability, allowing a model to perform satisfactorily even on an unpublished or unseen dataset. New studies are needed to investigate the feature representations learned by LeukNet, when compared to pre-trained models or even hand-crafted features. Future work may also investigate the use of Generative Adversarial Networks in increasing data availability, considering those are able to generate heterogeneous images that are sufficiently representative of the original distribution.

ACKNOWLEDGMENT

This study was financed in part by the "Coordenação de Aperfeiçoamento de Pessoal de Nível Superior" (CAPES), in Brazil, Finance Code 001, and by "Fundação de Amparo à Pesquisa do Piaui" (Fapepi). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

VI. PUBLICATIONS

- Rede Neural Convolucional para o Diagnóstico de Leucemia. Online version: https://sol.sbc.org.br/index.php/sbcas/article/view/6241.

- Diagnosis of Leukemia in Blood Slides based on a Fine-tuned and Highly Generalizable Deep Learning Model (Round 2 of reviews on Computers in Biology and Medicine).

REFERENCES

[1] M. Madhukar, S. Agaian, and A. T. Chronopoulos, "Automated screening system for acute myelogenous leukemia detection in blood microscopic images," *IEEE Systems Journal*, vol. 8, no. 3, pp. 995–1004, 2014.

[2] J. Yanas and E. Triantaphyllou, "A systematic survey of computer-aided diagnosis in medicine: Past and present developments," *Expert Systems with Applications*, vol. 138, p. 112821, December 2019.

[3] X. Li, L. Liu, J. Zhou, and C. Wang, "Heterogeneity analysis and diagnosis of complex diseases based on deep learning method," *Scientific Reports*, vol. 8, no. 6155, pp. 1–8, 2018.

[4] F. P. dos Santos and M. A. Ponti, "Alignment of local and global features from multiple layers of convolutional neural network for image classification," in *2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. IEEE, 2019, pp. 241–248.

[5] L. H. S. Vogado, R. M. S. Veras, F. H. D. Araújo, R. R. V. e Silva, and K. R. T. Aires, "Leukemia diagnosis in blood slides using transfer learning in cnns and SVM for classification," *Engineering Applications of Artificial Intelligence*, vol. 72, pp. 415–422, 2018.

[6] N. Patel and A. Mishra, "Automated leukaemia detection using microscopic images," *Procedia Computer Science*, vol. 58, pp. 635–642, 2015.

[7] V. Singhal and P. Singh, *Texture Features for the Detection of Acute Lymphoblastic Leukemia*. Singapore: Springer Singapore, 2016, vol. 409, pp. 535–543.

[8] T. T. P. Thanh, C. Vununu, S. Atoev, S.-H. Lee, and K.-R. Kwon, "Leukemia blood cell image classification using convolutional neural network," *International Journal of Computer Theory and Engineering*, vol. 10, no. 2, pp. 54–58, 2018.

[9] S. Shafique and S. Tehsin, "Acute lymphoblastic leukemia detection and classification of its subtypes using pretrained deep convolutional neural networks," *Technology in Cancer Research and Treatment*, vol. 17, pp. 1–7, september 2018.

[10] A. Rehman, N. Abbas, T. Saba, S. I. ur Rahman, Z. Mehmood, and H. Kolivand, "Classification of acute lymphoblastic leukemia using deep learning," *Microscopy Research and Technique*, pp. 1–8, October 2018.

[11] R. Sipes and D. Li, "Using convolutional neural networks for automated fine grained image classification of acute lymphoblastic leukemia," in *2018 3rd International Conference on Computational Intelligence and Applications (ICCIA)*, July 2018, pp. 157–161.

[12] T. Pansombut, S. Wikaisuksakul, K. Khongkraphan, and A. Phon-on, "Convolutional neural networks for recognition of lymphoblast cell images," *Computational Intelligence and Neuroscience*, vol. 2019, pp. 1–12, 2019.

[13] N. Ahmed, A. Yigit, Z. Isik, and A. Alpkocak, "Identification of leukemia subtypes from microscopic images using convolutional neural network," *Diagnostics*, vol. 9, no. 3, pp. 1–11, 2019.

[14] R. D. Labati, V. Piuri, and F. Scotti, "All-idb: The acute lymphoblastic leukemia image database for image processing." in *18th IEEE International Conference on Image Processing (ICIP)*, 2011, pp. 2045–2048.

[15] O. Sarrafzadeh and A. M. Dehnavi, "Nucleus and cytoplasm segmentation in microscopic images using k means clustering and region growing," *Advanced Biomedical Research*, pp. 79–87, December 2015.

[16] M. Rollins-Raval, J. Raval, and L. Contis, "Experience with cellavision dm96 for peripheral blood differentials in a large multi-center academic hospital system," *Journal of Pathology Informatics*, vol. 3, no. 29, pp. 1–9, 2012.

[17] A. T. H. U. B. Omid Sarrafzadeh, Hossein Rabbani, "Selection of the best features for leukocytes classification in blood smear microscopic images," in *Proc. SPIE*, vol. 9041, 2014, pp. 9041 – 9041 – 8.

[18] O. Sarrafzadeh, H. Rabbani, A. M. Dehnavi, and A. Talebi, "Detecting different sub-types of acute myelogenous leukemia using dictionary learning and sparse representation." in *ICIP*. IEEE, 2015, pp. 3339–3343.

[19] A. M. P. G. Vale, A. M. G. Guerreiro, A. D. D. Neto, G. B. Cavalvanti Junior, V. C. L. T. de Sá Leitão, and A. M. Martins, "Automatic segmentation and classification of blood components in microscopic images using a fuzzy approach," *Revista Brasileira de Engenharia Biomédica*, vol. 30, pp. 341–354, 2014.

[20] J. Böhm, "Pathologie-websites im world wide web," *Der Pathologe*, vol. 29, no. 3, pp. 231–242, 2008.

[21] X. Zheng, Y. Wang, G. Wang, and Z. Chen, "Fast and robust segmentation of white blood cell images by self-supervised learning," *Micron*, vol. 107, pp. 55–71, 2018.

[22] R. Duggal, A. Gupta, R. Gupta, and P. Mallick, "Sd-layer: Stain deconvolutional layer for cnns in medical microscopic imaging," in *Medical Image Computing and Computer Assisted Intervention MICCAI 2017*. Cham: Springer International Publishing, 2017, pp. 435–443.

[23] S. H. Rezatofighi and H. Soltanian-Zadeh, "Automatic recognition of five types of white blood cells in peripheral blood," *Computerized Medical Imaging and Graphics*, vol. 35, no. 4, pp. 333 – 343, 2011.

[24] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *CoRR*, vol. abs/1712.04621, 2017.

[25] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, July 2019. [Online]. Available: https://doi.org/10.1186/s40537-019-0197-0

[26] 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," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[27] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14, Cambridge, MA, USA, 2014, pp. 3320–3328.

[28] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional neural networks for medical image analysis: Full training or fine tuning?" *IEEE Transactions on Medical Imaging*, vol. 35, pp. 1299–1312, 2016.

[29] M. Izadyyazdanabadi, E. Belykh, M. Mooney, N. Martirosyan, J. Eschbacher, P. Nakaji, M. Preul, and Y. Yang, "Convolutional neural networks: Ensemble modeling, fine-tuning and unsupervised semantic localization for neurosurgical cle images," *Journal of Visual Communication and Image Representation*, vol. 54, pp. 10–20, 7 2018.

[30] F. H. Araujo, R. R. Silva, F. N. Medeiros, D. D. Parkinson, A. Hexemer, C. M. Carneiro, and D. M. Ushizima, "Reverse image search for scientific data within and beyond the visible spectrum," *Expert Systems with Applications*, vol. 109, pp. 35–48, 2018.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2016, pp. 770–778.

[33] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 2818–2826.

[34] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017, pp. 1800–1807.

[35] L. Zhang, L. Lu, I. Nogues, R. M. Summers, S. Liu, and J. Yao, "Deeppap: Deep convolutional networks for cervical cell classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 21, no. 6, pp. 1633–1643, 2017.

[36] A. Diaz-Pinto, S. Morales, V. Naranjo, T. Köhler, J. M. Mossi, and A. Navea, "Cnns for automatic glaucoma assessment using fundus images: an extensive validation," *BioMedical Engineering OnLine*, vol. 18, no. 29, pp. 1–19, 2019.

[37] E. Gibson, Y. Hu, H. J. Huisman, and D. C. Barratt, "Designing image segmentation studies: Statistical power, sample size and reference standard quality," *Medical Image Analysis*, vol. 42, pp. 44–59, 2017.

[38] S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?" in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 2661–2671.

[39] R. F. de Mello and M. A. Ponti, *Machine Learning: A Practical Approach on the Statistical Learning Theory*. Springer, 2018.