ACUTE LYMPHOBLASTIC LEUKEMIA DETECTION BASED ON ADAPTIVE UNSHARPENING AND DEEP LEARNING

Angelo Genovese, Mahdi S. Hosseini†, Vincenzo Piuri*, Konstantinos N. Plataniotis‡, Fabio Scotti**

* Department of Computer Science, Università degli Studi di Milano, Italy
† Department of Electrical & Computer Engineering, University of New Brunswick, Fredericton, NB, Canada
‡ Department of Electrical & Computer Engineering, University of Toronto, ON, Canada

ABSTRACT

Computer Aided Diagnosis (CAD) systems are increasingly utilizing image analysis and Deep Learning (DL) techniques, due to their high accuracy in several medical imaging fields, including the detection of Acute Lymphoblastic (or Lymphocytic) Leukemia (ALL) from peripheral blood samples. However, no method in the literature has specifically analyzed the focus quality of ALL images or proposed a technique for sharpening the samples in an adaptive way for the purpose of classification. To address this issue, in this paper we propose the first machine learning-based approach able to enhance blood sample images by an adaptive unsharpening method. The method uses image processing techniques and DL to normalize the radius of the cell, estimate the focus quality, adaptively improve the sharpness of the images, and then perform the classification. We evaluated the methodology on a public database of ALL images, considering several state-of-the-art CNNs to perform the classification, with results showing the validity of the proposed approach. *For a complete reproducibility of the work, the source code is available at:* `http://iebil.di.unimi.it/cnnALL/index.htm`.

***Index Terms*—** Deep Learning, CNN, ALL, XAI

1. INTRODUCTION

Acute Lymphoblastic (or Lymphocytic) Leukemia (ALL) is referred to a disease which affects the blood cells and rapidly spreads throughout the body, resulting in fatal consequences if left untreated. To ensure a timely detection of ALL (so to increase the chances of healing), an important step of the diagnosis is done by inspecting the white cells present in peripheral blood samples. The diagnosis is usually executed manually by an experienced pathologist who analyzes the malformations of white cells under the microscope, to determine the presence of cancer [1]. In particular, the pathologists examine peripheral blood looking for lymphoblasts, which are white cells with an altered morphology. Normally, lymphoblasts are present in the bone marrow, but an increased number of lymphoblasts in peripheral blood can be associated with ALL [2].

To help physicians in performing the diagnosis, there is an increasing interest in the development of Computer Aided Diagnosis (CAD) systems for ALL detection, which partially automate this process using image processing and Machine Learning (ML) techniques to detect lymphoblasts [3]. The associated methods can be divided in three main categories: *i)* handcrafted feature extraction and shallow ML classifiers; *ii)* handcrafted feature extraction and Deep Learning (DL); and *iii)* pure DL. The majority of the methods belong to the first category and process images using a handcrafted feature extraction step, then apply a shallow ML classifier (e.g., SVM) [4, 5, 6, 7, 8]. Methods with respect to the second category use a handcrafted feature extraction step, then apply a deep classifier (e.g., CNN) [9, 10, 11, 12]. Finally, methods in the third category use pure DL techniques without a handcrafted feature extraction step, taking advantage of their capability of automatically learning data representations. In some cases, these methods are able to detect lymphoblast cells with higher accuracy with respect to methods using handcrafted feature extraction [13, 14, 15, 16, 17, 18].

We acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research, within the project "Deep Learning and CUDA for advanced and less-constrained biometric systems".

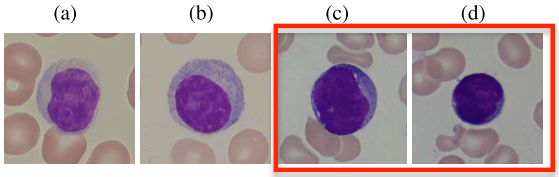

Fig. 1. Examples of white cells [1]: (a,b) *normal* white cells; (c,d) *lymphoblasts*.

Currently, the majority of DL-based methods for ALL detection are focusing on more efficient learning procedures [14, 16, 17] or original network architectures [13, 15, 18] to improve classification accuracy and detect lymphoblasts with more precision [3]. However, no method in the literature has specifically analyzed the focus quality of the samples or proposed an adaptive preprocessing method to sharpen the images with the purpose of classification.

In this paper we propose the first novel machine learning-based method based on image processing and DL for the focus quality estimation, adaptive unsharpening, and classification of ALL samples. The method uses novel adaptive image processing techniques to analyze the focus quality and improve the sharpness of the images before training the classifier. To tune the parameters of the unsharpening algorithm, the approach introduces the VAR-PCANet, a novel shallow CNN based on a feedforward design and trained using an unsupervised procedure, so that the size of the network and the filters only depend on statistical analyses of data in the previous layers. Lastly, the method uses state-of-the-art deep CNNs to process the unsharpened images and classify white blood cells in two classes: *normal* and *lymphoblast*, increasing the classification accuracy independently of which CNN is used. We evaluate our method on the Acute Lymphoblastic Leukemia Image Database (ALL-IDB) [1], with results showing that our approach can increase the lymphoblast detection accuracy, independently of which deep CNN is used in the final classification step.

The paper is structured as follows. Section 2 describes the methodology. Section 3 presents the experimental results. Finally, Section 4 concludes the work.

2. METHODOLOGY

The proposed method executes the following steps: *A)* image registration for normalizing cell radius; *B)* focus quality estimation and adaptive image unsharpening; *C)* shallow CNNs for tuning of adaptive image unsharpening; *D)* final adaptive image unsharpening;

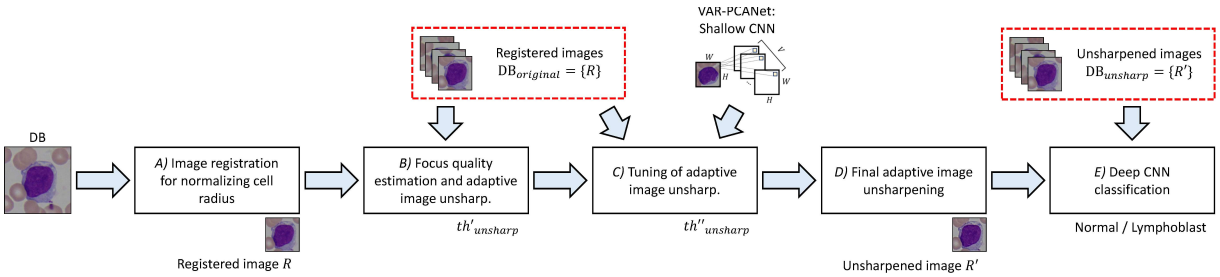

Fig. 2. Outline of the proposed methodology. In step E) "Deep CNN Classification", it is possible to choose among several pre-trained CNNs in the literature, such as AlexNet, VGG16, VGG19, ResNet, or DenseNet.

E) deep CNN classification. In the rest of the section, we consider a dataset containing images of white cells (Fig. 1) in which each image has a binary label (0: *normal*; 1: *lymphoblast*). Fig. 2 shows the outline of the proposed methodology.

2.1. Image Registration for Normalizing Cell Radius

The registration algorithm has the purpose of processing the dataset to normalize the radius of cells in each image.

First, we apply the color normalization method described in [19] to image I, convert the resulting image to grayscale, and binarize it using the Otsu's method, obtaining M_{thresh}.

Second, we segment the image by applying a fuzzy C-means clustering with 3 classes, and discarding the largest class corresponding to the background, obtaining the binary image M_{fcm}.

Third, the algorithm applies a post-processing by computing the union of the masks $M = M_{thresh} \cup M_{fcm}$, selecting only the connected component with the largest area, and refining the segmentation using the active contour method [20] followed by morphological processing.

Fourth, the method fits an ellipse on the edge coordinates of M, with center (c_x, c_y) and axes lengths a_{max}, a_{min}.

Lastly, we compute the registered image R by extracting the region of interest of I centered in (c_x, c_y), with size $1.5 \cdot a_{min}$, resize the resulting image to $W = H = 256$ pixels, and apply a min-max normalization to each image. In the following, we refer to the set of registered images $\{R\}$ as DB$_{original}$.

2.2. Focus Quality Estimation and Adaptive Image Unsharp.

An adaptive preprocessing technique is proposed to improve the sharpness quality of ALL images. The method is based on estimating the focus quality of each image, analyzing the relation among the focus quality and the image label, and on a subsequent processing to increase the sharpness quality in an adaptive way. To estimate the focus quality and perform the adaptive image unsharpening, we consider the training subset of the database.

We divide this adaptive approach into the following steps: *i)* estimation of focus quality; *ii)* estimation of data bias; *iii)* adaptive unsharpening.

2.2.1. Estimation of Focus Quality

The method estimates the focus quality f_i of each image R_i, with $1 \leq i \leq N$, where N is the number of training images in the database, obtaining the vector of focus qualities $\mathbf{f} = [f_1, f_2, \dots f_N]$. To compute the focus quality, we use the FQPath method introduced in [21], which evaluates the focus quality level of microscopy imaging in the brightfield mode by decomposing the input image using a visual sensitivity-like FIR filter corresponding to the out-of-focus lens, and then extracting high order statistical moment features to quantize the image sharpness level.

2.2.2. Estimation of Data Bias

The method extracts the vector of binary labels $\mathbf{l} = [l_1, l_2, \dots l_N]$, where $l_i = 1$ if R_i contains a lymphoblast, and $l_i = 0$ otherwise. Then, we estimate the data bias b by computing the correlation coefficient between $\mathbf{f}$ and $\mathbf{l}$: $b = \text{corrcoeff}(\mathbf{f}, \mathbf{l})$. We observed that, in the ALL-IDB database, there exist a significant data bias ($|b| > 50\%$), indicating that the focus quality is correlated with the label and the images with better quality are mostly associated with lymphoblast cells.

2.2.3. Adaptive Unsharpening

The approach performs an adaptive unsharpening of the images, with the purpose of improving the focus quality f_i for each image until it reaches $th_{unsharp}$, which represents a focus threshold. Such threshold is unique for the training subset and determines which focus quality the unsharpened images should have.

To perform the adaptive unsharpening, for each image $R_i \in \{R\}$, we compute the image R' by applying an unsharp masking using a Gaussian kernel with standard deviation σ_i. We chose the Gaussian kernel to perform the unsharp masking since such kernel simulates the optical characteristics of the out-of-focus lens [21]. To obtain a focus quality as close as possible to $th_{unsharp}$, σ_i is chosen separately for each image as the value minimizing the difference between the focus quality f_i and $th_{unsharp}$:

$$\sigma_i = \arg\min_{\sigma} \left(f_i - th_{unsharp} \right) \quad . \tag{1}$$

To choose the value of $th_{unsharp}$, we vary it in the range $[0, 10]$. Such range corresponds to the possible levels of image sharpness in medical imaging, (where $0 =$ "sharp" and $10 =$ "blurry"), provided by the FQPath focus quality estimation algorithm [21]. For each value, we compute the unsharpened images $\{R'\}$ and estimate the new focus quality f'_i. Then, we compute the vector of focus qualities $\mathbf{f}' = [f'_1, f'_2, \dots f'_N]$, and the new data bias $b' = \text{corrcoeff}(\mathbf{f}', \mathbf{l})$.

We obtain a value of b' for each possible value of $th_{unsharp}$. It is therefore possible to express the bias as a function of the focus threshold: $b' = f(th_{unsharp})$. Then, we choose the value of $th'_{unsharp}$ as the one that minimizes the absolute value of the data bias $b'(th_{unsharp})$:

$$th'_{unsharp} = \arg\min_{th_{unsharp}} \left(|b'(th_{unsharp})| \right) \quad . \tag{2}$$

2.3. Shallow CNNs for Tuning of Adaptive Image Unsharpening

This section describes the proposed method, based on shallow CNNs, to perform a machine learning-based tuning of the $th'_{unsharp}$ parameter of the adaptive image unsharpening algorithm. Such parameter is unique for the dataset and is computed considering all images of the training subset. In particular, we tune the $th'_{unsharp}$ parameter by training a shallow CNN on the unsharpened samples $\{R'\}$ obtained from the training subset, following the procedure described in Section 2.2.3. We consider different values of $th'_{unsharp}$

and choose the value for which we obtain the best classification accuracy.

In the proposed tuning method, we introduce the VAR-PCANet, a novel variation of the PCANet, a shallow CNN in which the filters are computed as the eigenvectors of the PCA on the input images [22]. We considered the PCANet since it has several properties that make it suitable to tune the proposed method for adaptive unsharpening: *i)* it has a feedforward design, in which the filters are derived from statistical analyses of the data in the previous layers; *ii)* it extracts a feature vector which can be analyzed to derive information about the distribution of the samples in the feature space. For example, for a new sample it would be possible to visualize the closest samples in the feature space; *iii)* it is possible to use a nearest neighbor classifier with no training, in which the output only depends on the feature vector and its closest sample in the feature space, to predict the label. In addition, PCANet represents a high-accuracy baseline in several computer vision tasks such as object recognition, biometrics, and medical image analysis [23, 24].

The VAR-PCANet extends the PCANet by providing an automatic methodology also to determine the sizing of the network. In the VAR-PCANet, the number of filters depends solely on the variance of the input data and is chosen using a learning procedure based on the PCA of the input images. In the rest of the section, we describe the procedure for the training of VAR-PCANet, feature extraction, classification, and parameter tuning. All the steps of the VAR-PCANet are applied considering the training subset of the database.

2.3.1. *Training of VAR-PCANet*

The training of the VAR-PCANet is divided in two steps as follows.

Collection of the Local Regions of the Images. After applying the adaptive image unsharpening, the method extracts from the images $\{R'\}$ the local regions $p_{i,j}$ with dimensions of $m_1 \times m_2$, centered on each pixel, with $i = 1, 2, \ldots, H$ and $j = 1, 2, \ldots, W$. Then, the method reshapes $p_{i,j}$ as $H \cdot W$ vectors $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{HW}$, with size $m_1 \times m_2$, and subtracts the mean value from each vector. We concatenate the vectors $\mathbf{p_i}$, computed from all the N images, to obtain a matrix P with size $m_1 m_2 \times NHW$.

Filter Tuning. The method performs the filter tuning using a PCA-based procedure. First, we select the V first eigenvectors of PP^T, as $\mathbf{e}_v$, with $v = 1, 2, \ldots, V$. Then, we select the V filters of the VAR-PCANet by reshaping each eigenvector $\mathbf{e}_v$ into a matrix with size $m_1 \times m_2$. Differently than the PCANet, in which V is chosen according to experimental evaluation, in the VAR-PCANet, V is chosen to preserve the th_{var} percentage of the input data, with $0 \leq th_{var} \leq 1$. In particular, we chose V as the number of filters for which the difference between the sum of the corresponding eigenvalues λ_v and th_{var} is minimum:

$$V = \arg\min_{V} \left(\left(\sum_{v=1}^{V} \lambda_v \right) - th_{var} \right) \quad , \tag{3}$$

where $\sum_{v=1}^{V} \lambda_v$ represents the percentage of the variance of the input data encoded by the first V eigenvectors.

2.3.2. *Feature Extraction and Classification*

First, we perform the feature extraction step by applying the trained VAR-PCANet on each image R' of the training subset and encoding the result as a feature vector, using a procedure based on image filtering, binary encoding, and histogram computation [22].

To classify the obtained feature vector, we use a k-NN classifier, with $k = 1$, based on the Euclidean distance (1-NN). We chose the 1-NN classifier since it does not require training nor parameters to tune and its accuracy only depends on the discriminant capability of the feature vectors. We consider the classification accuracy as the percentage of correctly classified samples.

2.3.3. *Parameter Tuning*

We tune the value of $th'_{unsharp}$ by following a procedure in two steps. First, we apply the method described in Section 2.2 to choose $th'_{unsharp}$ by minimizing the data bias $|b'|$, following the Eqn. 2. Then, we perform a sensitivity analysis by considering $th''_{unsharp} = th'_{unsharp} \pm 10\%$ and using the value of $th''_{unsharp}$ for which VAR-PCANet gives the best classification accuracy over the training subset, based on the procedure described in Section 2.3.2.

2.4. Final Adaptive Image Unsharpening

We apply the adaptive unsharpening method described in Section 2.2 separately on all images of the dataset (both training and testing subsets), using the obtained value of $th''_{unsharp}$. We refer to the set of preprocessed images $\{R'\}$ as $\text{DB}_{unsharp}$.

2.5. Deep CNN Classification

To perform the classification, we consider a pre-trained deep CNN, since they often represent the state of the art for classification in several fields, especially in the cases for which only a limited number of samples is available [14, 29]. To classify the blood samples, we perform a fine tuning of the deep CNN by modifying the last fully connected layer, which is configured for the 1000 classes of the ImageNet database, into a layer configured for a binary classification (0: *normal*; 1: *lymphoblast*). Then, we train the deep CNN on the training subset of the database, following the procedure described in Section 3.2. Lastly, we apply the trained deep CNN separately on each image of the testing subset to obtain the classification output.

3. EXPERIMENTAL RESULTS

For the experimental evaluation, we consider the ALL-IDB2 dataset [1], which contains 260 images of white cells cropped to show only a region of interest around the cell (Fig. 1). Each image has a binary label (0: *normal*; 1: *lymphoblast*).

3.1. Evaluation Procedure

To evaluate the accuracy of the proposed methodology, we consider a n-fold cross-validation, with $n = 2$, repeated 10 times. At each repetition, the training and testing subsets contain $\approx 50\%$ of the samples, selected randomly.

For each repetition, we apply the proposed methodology only on the training subset to perform the focus quality estimation and train the VAR-PCANet to tune the parameters of the adaptive image unsharpening algorithm. Then, we perform the final adaptive image unsharpening on all the dataset and classify the samples using the deep CNNs, trained using the procedure described in Section 3.2. The results are then averaged over the 10 repetitions.

3.2. Parameters Used and Deep CNN Training

We use a VAR-PCANet with 1 stage, with value $th_{var} = 0.92$, chosen by varying it in the range $[0, 1]$. We choose the values m_1, m_2 by considering the range $[1, W]$ and selecting the values corresponding to the best classification accuracy, obtaining $m_1 = m_2 = 15$.

We perform the fine tuning of each pre-trained deep CNNs by changing the last fully connected layer according to the procedure described in Section 2.5, then we train the deep CNN for 100 epochs, with a batch size $= 20$, using a learning rate of 1^{-4} for all layers, except for the last fully connected layer in which we use a learning rate of 2^{-3}. We use a data augmentation procedure on the training subset, by randomly flipping the image along the x and y axes and applying random rotations in the range of $[-180, 180]$ degrees.

Table 1. Quantitative evaluation: error measures on ALL-IDB2$_{original}$ and ALL-IDB2$_{unsharp}$, obtained using deep CNNs with fine tuning. For each of the CNNs separately, the ALL-IDB2$_{unsharp}$ database enables to obtain a greater classification accuracy.

Ref.	Deep CNN	Acc.	Err.	Sens.	Spec.	TP	TN	FP	FN
		(%) (Mean$_{\text{Std}}$)							
		ALL-IDB2$_{\text{original}}$							
[25]	AlexNet	$93.76_{2.06}$	$6.23_{2.06}$	$96.76_{2.10}$	$90.76_{4.52}$	$48.38_{1.05}$	$45.38_{2.26}$	$4.61_{2.26}$	$1.61_{1.05}$
[26]	VGG16	$95.30_{2.52}$	$4.69_{2.52}$	$96.76_{2.94}$	$93.84_{4.35}$	$48.38_{1.47}$	$46.92_{2.17}$	$3.07_{2.17}$	$1.61_{1.47}$
[26]	VGG19	$95.38_{2.05}$	$4.61_{2.05}$	$96.76_{1.52}$	$94.00_{4.88}$	$48.38_{0.76}$	$47.00_{2.47}$	$3.00_{2.47}$	$1.61_{0.76}$
[27]	ResNet18	$96.00_{1.01}$	$4.00_{1.01}$	$95.23_{2.45}$	$96.76_{1.84}$	$47.61_{1.22}$	$48.38_{0.92}$	$1.61_{0.92}$	$2.38_{1.22}$
[27]	ResNet50	$96.00_{1.48}$	$4.00_{1.48}$	$97.53_{1.80}$	$94.46_{2.82}$	$48.76_{0.90}$	$47.23_{1.41}$	$2.76_{1.41}$	$1.23_{0.90}$
[27]	ResNet101	$95.53_{1.97}$	$4.46_{1.97}$	$96.61_{1.89}$	$94.46_{4.17}$	$48.30_{0.94}$	$47.23_{2.08}$	$2.76_{2.08}$	$1.69_{0.94}$
[28]	DenseNet201	$96.76_{1.48}$	$3.23_{1.48}$	$95.23_{3.11}$	$98.30_{1.52}$	$47.61_{1.55}$	$49.15_{0.76}$	$0.84_{0.76}$	$2.38_{1.55}$
		ALL-IDB2$_{\text{unsharp}}$							
[25]	AlexNet	$95.07_{1.85}$	$4.92_{1.85}$	$96.92_{2.80}$	$93.23_{2.91}$	$48.46_{1.40}$	$46.61_{1.45}$	$3.38_{1.45}$	$1.53_{1.40}$
[26]	**VGG16**	$\mathbf{96.84_{1.27}}$	$\mathbf{3.15_{1.27}}$	$\mathbf{97.53_{1.80}}$	$\mathbf{96.15_{2.73}}$	$\mathbf{48.76_{0.90}}$	$\mathbf{48.07_{1.36}}$	$\mathbf{1.92_{1.36}}$	$\mathbf{1.23_{0.90}}$
[26]	VGG19	$95.53_{1.57}$	$4.46_{1.57}$	$94.76_{3.25}$	$96.30_{2.63}$	$47.38_{1.62}$	$48.15_{1.31}$	$1.84_{1.31}$	$2.61_{1.62}$
[27]	ResNet18	$96.00_{1.13}$	$4.00_{1.13}$	$94.76_{2.82}$	$97.23_{2.02}$	$47.38_{1.41}$	$48.61_{1.01}$	$1.38_{1.01}$	$2.61_{1.41}$
[27]	ResNet50	$96.69_{1.49}$	$3.30_{1.49}$	$97.38_{1.45}$	$96.00_{2.53}$	$48.69_{0.72}$	$48.00_{1.26}$	$2.00_{1.26}$	$1.30_{0.72}$
[27]	ResNet101	$96.00_{1.87}$	$4.00_{1.87}$	$95.84_{2.18}$	$96.15_{3.92}$	$47.92_{1.09}$	$48.07_{1.96}$	$1.92_{1.96}$	$2.07_{1.09}$
[28]	DenseNet201	$96.69_{1.14}$	$3.30_{1.14}$	$95.07_{2.27}$	$98.30_{2.34}$	$47.53_{1.13}$	$49.15_{1.17}$	$0.84_{1.17}$	$2.46_{1.13}$

Notes. Acc. = Accuracy; Err. = Error; Sens. = Sensitivity; Spec. = Specificity; TP = True Positives; TN = True Negatives; FP = False Positives; FN = False Negatives.

3.3. Quantitative Evaluation

First, we perform a fine tuning of the deep CNNs on ALL-IDB2$_{original}$ and ALL-IDB2$_{unsharp}$ separately, we perform the classification, and compute the error measures, as described in Section 3.1. As CNNs, we compare the results obtained using AlexNet [25], VGG16, VGG19 [26], ResNet [27], and DenseNet [28]. These CNNs are among the most widely used in the literature [29]. Table 1 shows the error measures obtained using different deep CNNs with fine tuning, on both ALL-IDB2$_{original}$ and ALL-IDB2$_{unsharp}$. For all methods, we report the mean and standard deviation, following the reporting procedure described in [1]. From the table, it is possible to observe that, for each CNN separately, using ALL-IDB2$_{unsharp}$ enables to obtain a greater classification accuracy and therefore a lower classification error, showing the validity of the proposed methodology. In particular, using the VGG16 fine tuned on ALL-IDB2$_{unsharp}$ permits to obtain the greatest classification accuracy among the considered methods ($\mathbf{96.84}\%$).

As a comparison, we considered a traditional unsharpening method, with a fixed value of σ_i for all images, chosen in the range $[1, 5]$. Beyond this range, the corresponding focus quality level f_i did not change significantly. However, a fixed value of σ_i did not allow to increase the classification accuracy in all cases. Moreover, the optimal value of σ_i would need to be chosen separately for each CNN using an experimental procedure. In contrast, our method automatically tunes σ_i using a procedure based on machine learning.

3.4. Qualitative Evaluation

To perform a qualitative evaluation of the proposed methodology, we apply the Grad-CAM technique [30] on the VGG16 trained with fine tuning on the ALL-IDB2$_{unsharp}$, as described in Section 2.5. We chose the Grad-CAM since it allows to observe which regions of the image contribute more to the result of the classification. With respect to other similar methods such as LIME [31] or SHAP [32], Grad-CAM can output a finer-grained map describing how each region of the image contributes to the final classification.

To evaluate in an qualitative way the improvement in the accuracy obtained using ALL-IDB2$_{unsharp}$ with respect to ALL-IDB2$_{original}$, we extract the pairs of corresponding images (R_i, R'_i), where $R_i \in$ ALL-IDB2$_{original}$ and $R'_i \in$ ALL-IDB2$_{unsharp}$, with $i = 1, \ldots, M$, where M is the size of the dataset. Then, we compare the results of the Grad-CAM on R_i and R'_i, for each i. Fig. 3 shows some examples of the Grad-CAM applied on images of ALL-IDB2$_{original}$ and ALL-IDB2$_{unsharp}$. The regions in red correspond to the parts of the image that contribute more to the classification output. From the figures, it is possible to observe how, in the images $R'_i \in$ ALL-IDB2$_{unsharp}$, the regions with higher intensity (in red) are more centered on the white cells, with respect to the images $R_i \in$ ALL-IDB2$_{original}$. This is caused by the fact that, in the case of ALL-IDB2$_{unsharp}$, the features of the cell have a greater influence on the result of the classification, with respect to the images in ALL-IDB2$_{original}$. This simply implies how the proposed method is able to increase the accuracy of the classification by increasing the level of detail and allows CNNs to learn features focused more on the details on the cell, and not on the background.

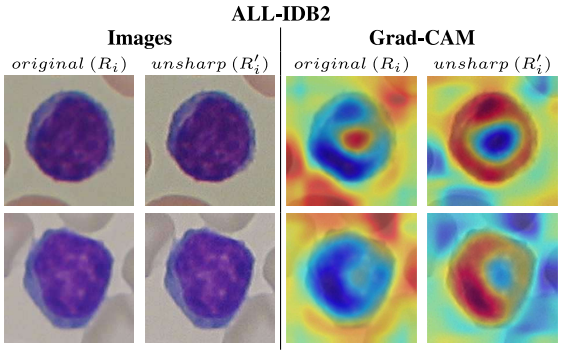

Fig. 3. Qualitative evaluation: examples of corresponding images of ALL-IDB2$_{original}$ (left column) and ALL-IDB2$_{unsharp}$ (right column), with the application of the Grad-CAM technique [30] on the results obtained using VGG16 with fine tuning. It is possible to observe that, in the images in ALL-IDB2$_{unsharp}$, the regions that contribute more to the classification output (in red) are more centered on the white cells, with respect to using ALL-IDB2$_{original}$, showing that proposed method increases the level of detail and allows CNNs to learn features focused more on the details on the cell.

4. CONCLUSION

In this paper, we proposed the first machine learning-based methodology based on Deep Learning (DL) for the focus quality estimation, adaptive unsharpening, and classification of Acute Lymphoblastic Leukemia (ALL) blood samples as *normal* vs *lymphoblast*. Our method uses innovative adaptive image processing techniques to improve the image sharpness level prior to training and shallow CNNs to tune the parameters of the unsharpening algorithm. The proposed approach increases the details of the images by estimating their focus quality and adaptively reducing the bias in the data between the quality of the images and their class. Experiments on public ALL databases show that deep CNNs trained using the images unsharpened with the proposed method increase the lymphoblast detection accuracy, independently of which CNN is used. Future works should consider different DL architectures and databases with more samples.

5. REFERENCES

[1] R. Donida Labati, V. Piuri, and F. Scotti, "ALL-IDB: The Acute Lymphoblastic Leukemia Image Database for image processing," in *Proc. of ICIP*, 2011.

[2] M. M. Amin, S. Kermani, A. Talebi, and M. G. Oghli, "Recognition of acute lymphoblastic leukemia cells in microscopic images using k-means clustering and support vector machine classifier," *J. Medical Signals Sens.*, vol. 5, no. 1, Jan. 2015.

[3] H. T. Salah, I. N. Muhsen, M. E. Salama, T. Owaidah, and S. K. Hashmi, "Machine learning applications in the diagnosis of leukemia: Current trends and future directions," *Int. J. Lab. Hematol.*, vol. 41, no. 6, Dec. 2019.

[4] S. Mishra, B. Majhi, P. K. Sa, and L. Sharma, "Gray level co-occurrence matrix and random forest based acute lymphoblastic leukemia detection," *Biomed. Signal Proces.*, vol. 33, Mar. 2017.

[5] J. Rawat, A. Singh, H. S. Bhadauria, J. Virmani, and J. S. Devgun, "Classification of acute lymphoblastic leukaemia using hybrid hierarchical classifiers," *Multimed. Tools Appl.*, vol. 76, no. 18, Sep. 2017.

[6] W. Srisukkham, L. Zhang, S. C. Neoh, S. Todryk, and C. P. Lim, "Intelligent leukaemia diagnosis with bare-bones PSO based feature optimization," *Appl. Soft Comput.*, vol. 56, Jul. 2017.

[7] M. MoradiAmin, A. Memari, N. Samadzadehaghdam, S. Kermani, and A. Talebi, "Computer aided detection and classification of acute lymphoblastic leukemia cell subtypes based on microscopic image analysis," *Microsc. Res. Tech.*, vol. 79, no. 10, Oct. 2016.

[8] S. C. Neoh, W. Srisukkham, L. Zhang, S. Todryk, B. Greystoke, C. P. Lim, M. A. Hossain, and N. Aslam, "An intelligent decision support system for leukaemia diagnosis using microscopic blood images," *Sci. Rep.*, vol. 5, no. 14938, Oct. 2015.

[9] K. K. Jha and H. S. Dutta, "Mutual information based hybrid model and Deep Learning for Acute Lymphocytic Leukemia detection in single cell blood smear images," *Comput. Methods Programs Biomed.*, vol. 179, Oct. 2019.

[10] J. Laosai and K. Chamnongthai, "Deep-Learning-based Acute Leukemia classification using imaging flow cytometry and morphology," in *Proc. of ISPACS*, 2018.

[11] J. Zhao, M. Zhang, Z. Zhou, J. Chu, and F. Cao, "Automatic detection and classification of leukocytes using convolutional neural networks," *Med. Biol. Eng. Comput.*, vol. 55, no. 8, Aug. 2017.

[12] M. I. Razzak and S. Naz, "Microscopic blood smear segmentation and classification using deep contour aware CNN and Extreme Machine Learning," in *Proc. of CVPRW*, 2017.

[13] R. Duggal, A. Gupta, R. Gupta, and P. Mallick, "SD-Layer: Stain deconvolutional layer for CNNs in medical microscopic imaging," in *Proc. of MICCAI*, 2017.

[14] S. Shafique and S. Tehsin, "Acute Lymphoblastic Leukemia detection and classification of its subtypes using pretrained Deep Convolutional Neural Networks," *Technol. Cancer Res. T.*, vol. 17, Jan. 2018.

[15] J. L. Wang, A. Y. Li, M. Huang, A. K. Ibrahim, H. Zhuang, and A. M. Ali, "Classification of white blood cells with PatternNet-fused ensemble of Convolutional Neural Networks (PECNN)," in *Proc. of ISSPIT*, 2018.

[16] A. Rehman, N. Abbas, T. Saba, S. I. u. Rahman, Z. Mehmood, and H. Kolivand, "Classification of Acute Lymphoblastic Leukemia using Deep Learning," *Microsc. Res. and Tech.*, vol. 81, no. 11, Nov. 2018.

[17] A. Talaat, P. Kollmannsberger, and A. Ewees, "Efficient classification of white blood cell leukemia with improved swarm optimization of deep features," *Scientific Reports*, vol. 10, no. 2536, Feb. 2020.

[18] P. Mathur, M. Piplani, R. Sawhney, A. Jindal, and R. R. Shah, "Mixup multi-attention multi-tasking model for early-stage leukemia identification," in *Proc. of ICASSP*, 2020.

[19] G. D. Finlayson, B. Schiele, and J. L. Crowley, "Comprehensive colour image normalization," in *Proc. of ECCV*, 1998.

[20] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE T. Image Process.*, vol. 10, no. 2, Feb. 2001.

[21] M. S. Hosseini, J. A. Z. Brawley-Hayes, Y. Zhang, L. Chan, K. N. Plataniotis, and S. Damaskinos, "Focus quality assessment of high-throughput whole slide imaging in digital pathology," *IEEE T. Med. Imaging*, vol. 39, no. 1, Jan. 2020.

[22] T. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: A simple Deep Learning baseline for image classification?," *IEEE T. Image Process.*, vol. 24, no. 12, Dec. 2015.

[23] J. Shi, J. Wu, Y. Li, Q. Zhang, and S. Ying, "Histopathological image classification with color pattern random binary hashing-based PCANet and matrix-form classifier," *IEEE J. Biomed. Health*, vol. 21, no. 5, Sep. 2017.

[24] A. Genovese, V. Piuri, K. N. Plataniotis, and F. Scotti, "PalmNet: Gabor-PCA Convolutional Networks for touchless palmprint recognition," *IEEE T. Inf. Foren. Sec.*, vol. 14, no. 12, Dec. 2019.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. of NIPS*, 2012.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. of ICLR*, 2015.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of CVPR*, 2016.

[28] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected Convolutional Networks," in *Proc. of CVPR*, 2017.

[29] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep Learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, Feb. 2020.

[30] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from Deep Networks via gradient-based localization," in *Proc. of ICCV*, 2017.

[31] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you?: Explaining the predictions of any classifier," in *Proc. of KDD*, 2016.

[32] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. of NIPS*, 2017.