Image and Vision Computing 151 (2024) 105298

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

A decision support system for acute lymphoblastic leukemia detection based on explainable artificial intelligence

Angelo Genovese *, Vincenzo Piuri, Fabio Scotti

Department of Computer Science, Università degli Studi di Milano, Italy

ARTICLE INFO

Keywords:
Acute lymphoblastic leukemia (ALL)
Explainable artificial intelligence (XAI)
Deep learning (DL)
Convolutional neural network (CNN)

ABSTRACT

The detection of acute lymphoblastic leukemia (ALL) via deep learning (DL) has received great interest because of its high accuracy in detecting lymphoblasts without the need for handcrafted feature extraction. However, current DL models, such as convolutional neural networks and vision Transformers, are extremely complex, making them black boxes that perform classification in an obscure way. To compensate for this and increase the explainability of the decisions made by such methods, in this paper, we propose an innovative decision support system for ALL detection that is based on DL and explainable artificial intelligence (XAI). Our approach first introduces causality into the decision with a metric learning approach, enabling a decision to be made by analyzing the most similar images in the database. Second, our method integrates XAI techniques to allow even non-trained personnel to obtain an informed decision by analyzing which regions of the images are most similar and how the samples are organized in the latent space. The results on publicly available ALL databases confirm the validity of our approach in opening the black box while achieving similar or superior accuracy to that of existing approaches.

1. Introduction

Acute lymphoblastic (or lymphocytic) leukemia (ALL) is a disease that affects white blood cells (WBCs) by modifying their morphology from a "normal" state to an "altered" state. WBCs with this altered morphology are called lymphoblasts (Fig. 1); they can be present in small numbers even in healthy individuals, especially in the bone marrow. However, an increased number of lymphoblasts in peripheral blood can indicate the presence of ALL [1,2].

The traditional approach to detecting ALL is to perform manual inspection of samples of WBCs and check the presence and number of lymphoblasts in the blood. However, such inspection is time consuming and prone to errors and can lead to fatigue. Moreover, the availability of an experienced pathologist is not always guaranteed [3]. To overcome the disadvantages of manual inspection, which can be performed only by trained personnel, computer-aided diagnosis (CAD) systems based on artificial intelligence (AI) have been increasingly researched to help pathologists make diagnoses by performing a preliminary classification of blood samples and detecting possible lymphoblasts [4]. In particular, AI-based approaches using deep learning (DL) have shown remarkable accuracy in medical imaging [5,6], digital pathology [7], and ALL detection [8].

DL methods for ALL detection have exhibited high detection accuracy on several publicly available datasets of ALL images; DL-based models such as convolutional neural networks (CNNs) [9] and vision Transformers (ViTs) [10] have been applied. However, the extremely high number of complex parameters involved in these models make them black boxes, which have the drawback that the decision is often made in an obscure way [11].

To "open" the black box and provide information that can help humans understand DL models, explainable artificial intelligence (XAI) methods have been proposed in the literature, both general and application-specific [12], and some of these approaches (e.g., CAM-based methods) have been routinely used along with accuracy measures to show the quality of the training [13]. In fact, XAI techniques can produce more trustable models, important to ensure fair decision and also to reduce the possibility of adversarial attacks. However, no XAI method for ALL detection has considered a causability approach, where the classification is supported by information that is useful for the expert in determining whether to trust the decision (e.g., by providing the cause of a specific classification) [11]. Moreover, no paper in the literature for ALL detection has yet integrated XAI techniques into a decision

* Corresponding author.
E-mail address: angelo.genovese@unimi.it (A. Genovese).

https://doi.org/10.1016/j.imavis.2024.105298
Received 31 January 2024; Received in revised form 23 September 2024; Accepted 7 October 2024
Available online 11 October 2024

support system, where the outcome of the XAI approach not only indicates high model generalizability but can also be effectively used to make an informed decision.

In this paper, we propose an innovative decision support system for ALL detection that is based on DL and XAI,[1] with the goal of *i)* introducing causability and *ii)* integrating XAI techniques to make an informed decision. To introduce causability into the explanation, we first train a model via a metric learning approach, which organizes the latent space by keeping samples in the same class close to each other and farther away from samples in other classes. Second, in the test phase, for each query sample, we retrieve the training images that are closer to it in the latent space and then classify the query sample on the basis of the labels of the close retrieved images. Therefore, we bring causability into the classification decision by associating each sample with the images that are most similar to it, for which we already know the label.[2]

To integrate XAI techniques into the decision support system, we combine CAM-based methods to highlight both which parts of the images contributed to the decision and also which parts of the images make them similar to each query sample. We also integrate dimensionality reduction methods in a way that highlights the position of the query sample with respect to the close images in the reduced latent space, providing additional causability information in support of the decision. Some of the techniques proposed in this work were also considered in the paper described in [10], which proposed novel transfer learning algorithms based on both CNNs and the ViT to classify ALL samples accurately, as well as XAI techniques based on CAM and dimensionality reduction. However, the authors of that study considered XAI techniques only to evaluate the quality of the training and did not use them to bring causability into the decision. Moreover, that study did not consider any method of organizing the latent space, and classification was performed in an obscure way. In contrast, in our work, we introduce metric learning and integrate XAI techniques into a decision support system that brings causability into the decision.

This work is the first method in the literature to propose a decision support system for ALL detection that integrates metric learning, majority voting, and XAI techniques to perform accurate and explainable classification while bringing causability and insight into the decision to increase confidence in the decision. Our goal is to provide the reasoning behind the classification by aggregating the labels and showing the images obtained from the most similar samples. While research on DL-based ALL detection is increasing the number of methods, accuracy, and number of available datasets [14], deployment of the proposed methods in clinical practice is still hindered by several factors, including limited consensus about what constitutes an explainable decision [15]. Hopefully, our contribution will pave the way for greater acceptance of CAD systems based on DL.

The remainder of the paper is structured as follows. Section 2 presents the literature review. Section 3 introduces the method used to train the model. Section 4 describes the decision support system used to perform classification. Section 5 discusses the experimental results. Finally, Section 6 concludes the paper.

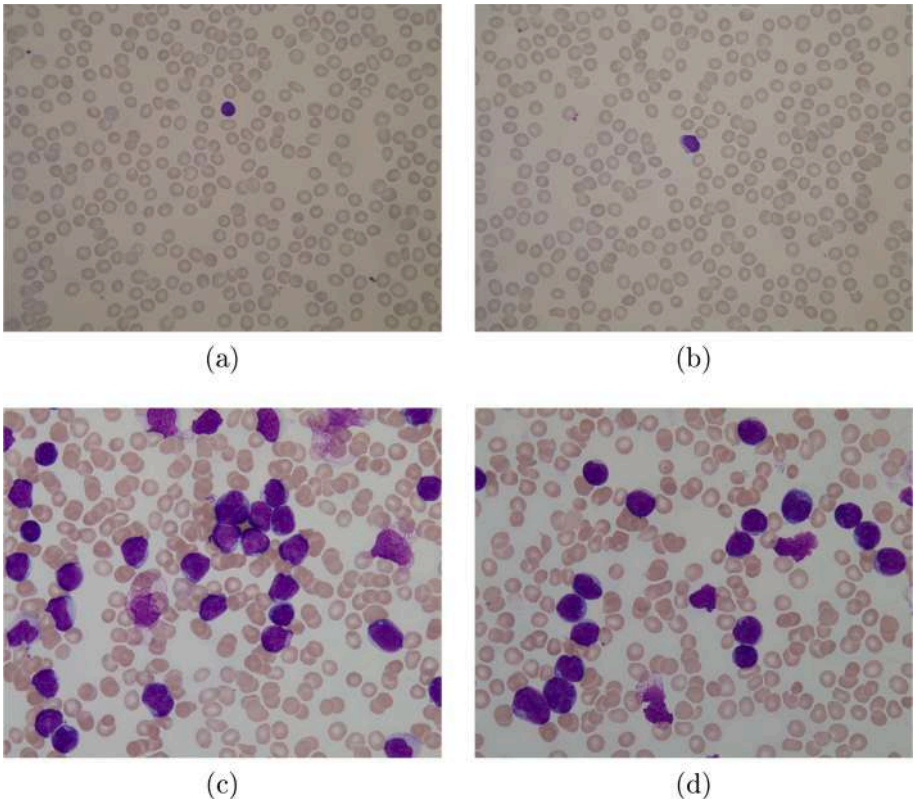

Fig. 1. Examples of images of white blood cells (WBCs) taken from different subjects [1]: *(a, b) normal* WBCs in a healthy subject; *(c, d) lymphoblasts* in a subject affected by ALL. The images are stained via the hematoxylin and eosin technique and the WBCs are visible in purple. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2. Related works

The traditional methods for ALL detection that used AI techniques

[1] The source code is available at https://github.com/AngeloUNIMI/DSS_XAI_ALL

[2] Colloquially speaking, our decision support system answers the classification question for an unknown sample by saying: *I classify this sample with the label 'x' because it is most similar to these images, which also have the label 'x'.*

were often based on a handcrafted feature extraction step followed by a shallow classifier [16,17]. While such methods rely on features that are more explainable, they are also limited by features that are fixed and chosen by nontrained personnel (e.g., computer scientists rather than pathologists). To address the limitations of handcrafted features, DL-based approaches are currently predominantly used [8] since they have the ability to automatically learn feature representations [18]. This ability, in turn, reduces the need for pathologists to be involved in the design process (except by labeling the samples [1]) and enables higher classification accuracy, provided that sufficient training data are available [10].

DL-based methods for ALL detection can be divided into three categories on the basis of the approach used to achieve a higher classification accuracy [19]: *i)* original learning procedures; *ii)* ad hoc network architectures; *iii)* DL-based preprocessing.

Methods belonging to the first category focus on improving the training algorithm to achieve a more accurate classification of ALL samples. Such approaches include pretraining the CNN on a larger database (e.g., ImageNet) and then fine-tuning it on an ALL database to achieve higher accuracy [20–23], following a standard procedure for medical imaging in the case of small databases, for which training a CNN from scratch would result in overfitting the training data and therefore a lower accuracy. [24]. Similarly, the approaches described in [19,25] perform pretraining, but this is done on a histopathology database, leveraging the fact that the images are more similar to those in the target domain. Additionally, by leveraging the similarity of histopathology images to ALL samples, the approach presented in [10] obtains a multitask DL model trained via cross-dataset transfer learning, which considers both the source and target databases at the same time. This work considers XAI techniques that are based on CAM and dimensionality reduction; however, the purpose is only to evaluate the quality of the training.

Unlike approaches that use only pretraining followed by fine tuning, the method introduced in [26] considers a swarm optimization approach after the pretraining step, with the goal of selecting the features that are more suitable for ALL classification. In addition to considering the ALL database as having a limited dimensionality, [27] describes the best approaches for dealing with class imbalance, for example, when images from healthy subjects are more widely available than images from subjects affected by ALL.

Methods belonging to the second category describe modifications of standard DL architectures that aim to extract more discriminant feature representations from blood images. Examples include the approach presented in [28], where a variation of the ResNet CNN architecture was used to capture details of blood samples both at local and global scales, and the method introduced in [29], where longer skip connections limit the problem of the vanishing gradient and more strongly affect shallower layers in the CNN. Other approaches have introduced ad hoc architectures with the main purpose of avoiding overfitting on small-sized ALL datasets. Such methods include the approach described in [30], which uses Bayesian CNNs; the method presented in [31], which proposes a shallow ResNet variant; and the method proposed in [32], which is based on a CNN with fixed weights. Similarly, other approaches have considered shallow or optimized CNN architectures by focusing on computational complexity, such as the methods described in [33,34]. Finally, to increase the reliability of the prediction, several methods have used architectures that can process multiple images at the same time [35,36].

Methods belonging to the third category use DL-based models specifically designed to enhance blood images and therefore improve the classification accuracy. Such approaches include that of [37], which uses a convolutional layer designed specifically for blood images by performing stain deconvolution, and the method presented in [38], which uses a shallow CNN to tune an unsharpening algorithm and increase the focus quality of the images.

To our knowledge, no paper in the literature has considered a metric learning approach to bring causability into the decision or a decision support system based on XAI for ALL detection.

3. Model training methodology

This section describes the methodology for training the model, which is used in the decision support system to perform classification. Fig. 2 shows the outline of the methodology.

Step A: We consider histopathological pretraining of the model to leverage a larger and more general-purpose database that includes pathological images.

Step B: We perform class injection on the ALL database by including images with no WBCs at all (by adding *ALL class* 0) and splitting the images of the "lymphoblast" class into separate classes (by adding *ALL class* 3). This is motivated by the idea that if images in the same class have distinct appearances, this hinders training.

Step C: We fine-tune the model on the ALL database with a metric learning approach, which leverages the greater separability of the classes achieved by the class injection step.

3.1. Multitask histopathological pretraining

We perform multitask histopathological pretraining in two steps. First, we take a standard model architecture (e.g., ResNet) and substitute the classification layer (the last fully connected (FC) layer) with a multitask classification layer consisting of n_{FC} FC layers in parallel (Fig. 3). In particular, we consider n_{FC} FC layers to create a multi-task learning structure and allow training the model using databases that contain labels with different cardinality. By adding multiple FC layers in parallel, it is possible to consider all labels during training, which can result in the model learning more general features and thus being more transferable [39].

Second, we train the resulting model on a database of histopathological images. The database consists of patches extracted from whole-slide images (WSIs), with each patch depicting different tissues of the human body, such as adipose, nervous, skeletal, or epithelial tissue. Each patch has a corresponding label describing which tissue is present in the patch; each patch can contain multiple tissues, and therefore, the labels are not mutually exclusive. In particular, the database considered in our study contains $n_{FC} = 3$ different levels of labels, organized in a hierarchical way: each level describes a more precise label of the patch [40].

To train the model, we compute the L_j loss for the j-th FC layer, with $1 < j < 3$, by comparing the output of the FC layer with the corresponding level of labeling, using a multilabel soft margin loss:

$$L_j = -\frac{1}{C}\sum_i \mathbf{w}_j[i] \quad \mathbf{y}_j[i] log\Big((1 + exp(-\mathbf{x}[i]))^{-1}\Big) + \Big(1 - \mathbf{y}_j[i]\Big) log\left(\frac{exp(-\mathbf{x}[i])}{(1 + exp(-\mathbf{x}[i]))}\right), \tag{1}$$

where $\mathbf{y}_j$ is the label vector for the j-th level of labeling, C is the cardinality of $\mathbf{y}_j$, and $\mathbf{w}_j$ is the vector of class weights. Each weight $w[i] \in \mathbf{w}_j$ corresponds to a tissue type and is computed as the inverse of the percentage of the cardinality of the tissue type to the total number of samples. Multilabel soft margin loss is chosen because the samples in the histopathological database each have multiple labels.

Then, we aggregate the L_j losses. When training a CNN, we also consider the L_{orth} orthogonality loss, which requires that the kernels of the CNN be as orthogonal as possible, thereby reducing redundancy [25,41]. As a result, we obtain the $L_{pretrain}$ loss:

$$L_{pretrain} = \left(\frac{1}{3}\sum_{j=1}^{3} L_j\right) + \lambda L_{orth}, \tag{2}$$

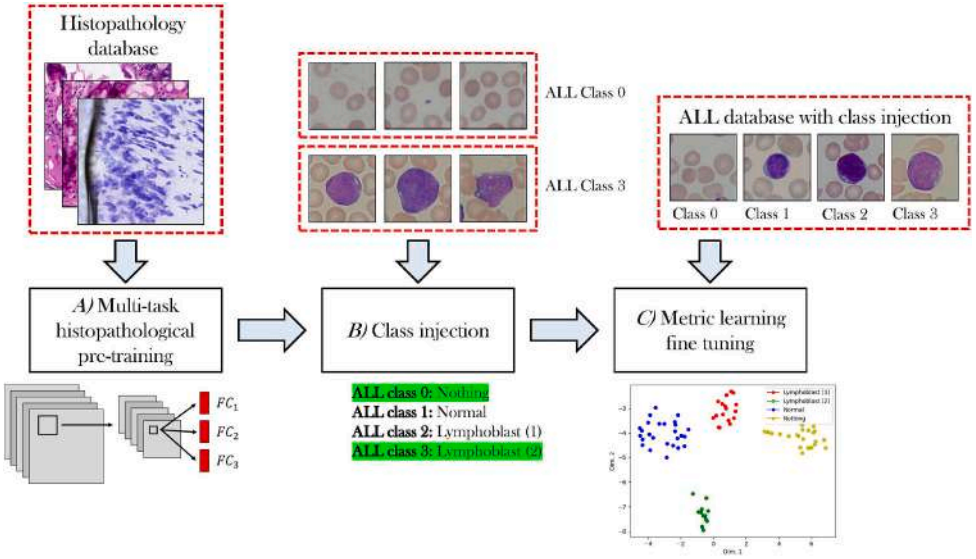

Fig. 2. Outline of the proposed methodology for training the model, which is used in the decision support system. The following steps are performed: *A)* multitask histopathological pretraining, in which we train the model on a histopathology database; *B)* class injection, in which we add images with no WBCs (by adding *ALL class* 0) and split the lymphoblast class (by adding *ALL class* 3); *C)* metric learning fine tuning, in which we tune the model on ALL images via metric learning.

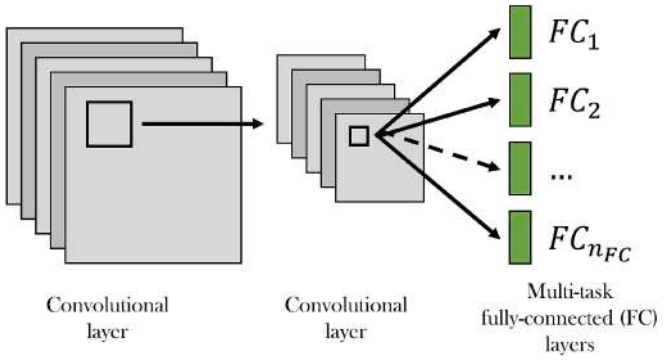

Fig. 3. The multitask architecture used for performing histopathological pretraining. Instead of a single fully connected (FC) layer, we consider a multitask architecture with n_{FC} FC layers in parallel. For simplicity, we consider a CNN in this figure.

where $0 < \lambda < 1$ is the weight of the orthogonality loss.

3.2. Class injection

We perform class injection in two steps. First, we inject images that do not contain any WBCs. In fact, most ALL databases have only images of "normal" WBCs (healthy cases) and "lymphoblasts" (possible ALL cases). Therefore, we complement the database by adding images with no WBCs (*ALL class* 0). Since in most images of WBCs, there are several red blood cells in the background, in this work, we consider images in which only red blood cells are visible, which are taken from WSIs of blood samples [25].

The injection of images in ALL class 0 forces the learning algorithm to detect the presence of normal WBCs, which enables the model to learn features that are distinctive of "normal" WBCs. In a two-class database (1: *normal*; 2: *lymphoblasts*), without the presence of ALL class 0, a model would only need to differentiate lymphoblasts from "anything else", and thus, it would not need to learn any discriminating features for normal WBCs. In contrast, in our approach, the learned features for normal WBCs are important when analyzing the result via XAI methods, as they make it possible to understand what parts of the image contributed to the model classifying a WBC as "normal".

Second, we inject additional classes by splitting images with the same label into different classes on the basis of the analysis of the features learned. In fact, some databases might contain images captured at different magnifications or resolutions that nonetheless have the same label (Fig. 4). When a classification model is applied, the features of images captured in different ways present a distinctive bimodal distribution, with the corresponding features clustered in different positions of the latent space. To facilitate metric learning, we split such images into different labels and inject additional classes.

Fig. 5 shows an example of the t-SNE dimensionality reduction method applied to the features obtained by a trained model on an ALL database with the abovementioned issue, before and after the injection of additional classes. The figure shows that the "lymphoblast" class was originally bimodal, and injecting the additional class "lymphoblast (2)" enabled neater separation of the classes.

3.3. Metric learning fine-tuning

We fine-tune the model, which is obtained after the multitask histopathological pretraining described in Section 3.1, on the ALL database processed via the class injection procedure described in Section 3.2.

As the loss function, we consider a metric learning loss L_{metric}. In particular, we consider the triplet margin loss,[3] which processes three tensors: the anchor a, the positive p, and the negative n:

$$L_{metric}(a,p,n) = max\{d(a_i,p_i) - d(a_i,n_i) + 1\,,0\}, \tag{3}$$

where $d = \|x_i - y_i\|_2$. The purpose of L_{metric} is to bring the values of a and p closer together while separating the values of a and n. As the values of a,p,n, we choose the outputs of the model before the last FC layer for two samples of the same class a,p and one sample of a different class n. As a consequence, L_{metric} tends to organize and cluster the latent space so that samples of the same class are closer to each other. Then, we add to L_{metric} the L_{orth} orthogonality loss and the cross-entropy loss L_{CE} to distinguish the different classes:

[3] https://github.com/KevinMusgrave/pytorch-metric-learning

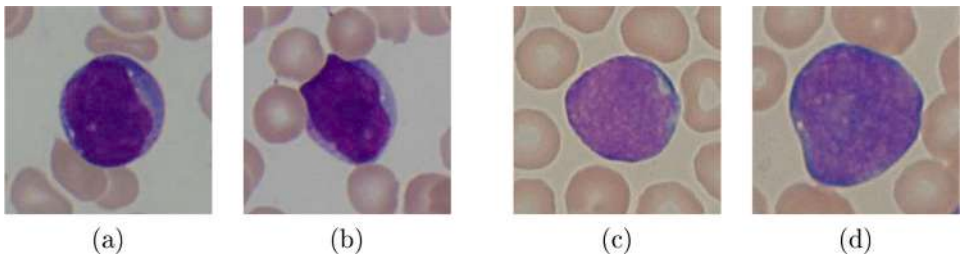

Fig. 4. Examples of images from an ALL database captured with different magnifications or resolutions that have the same label: *(a, b)* first magnification; *(c, d)* second magnification. It is possible that the model will learn very different features for the two magnifications, even if the labels are the same. To address this problem, we perform class injection and separate the images into two different labels.

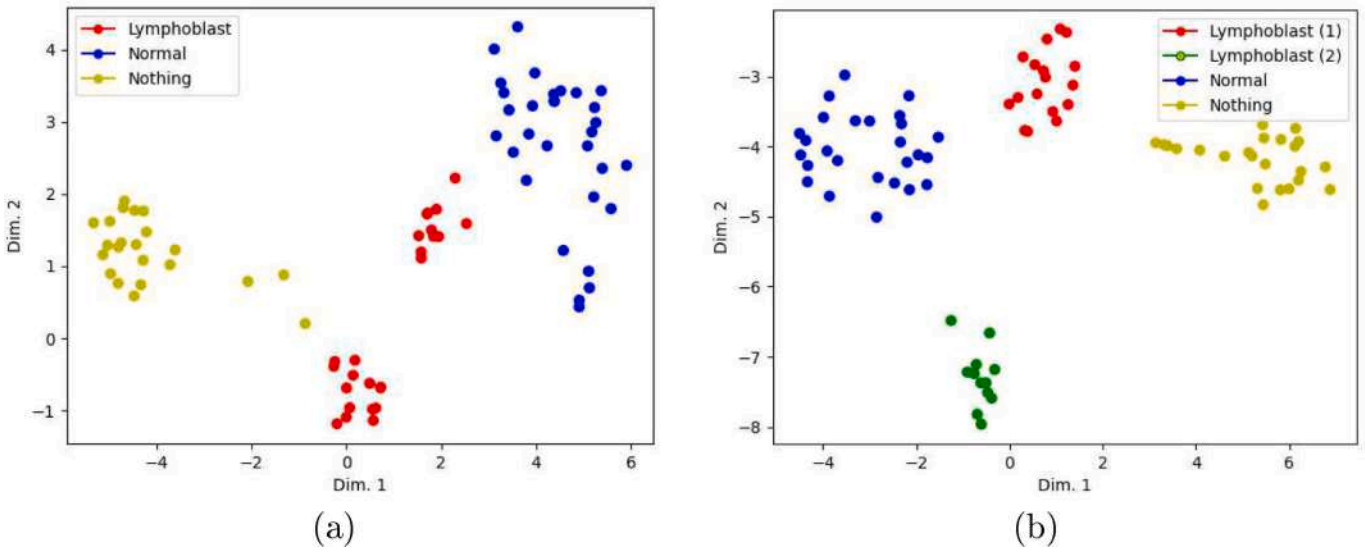

Fig. 5. Example of t-SNE dimensionality reduction applied to an ALL database: *(a)* before the injection of an additional "lymphoblast" class and *(b)* after the injection. In *(a)*, the "lymphoblast" class is bimodal. This is probably caused by images with the same label that were captured at different magnifications or resolutions. To facilitate metric learning, we split such images into different labels and inject an additional "lymphoblast" class. In *(b)*, the result of t-SNE shows a neater separation of the classes.

$$L_{CE}(x, class) = -w[i] \quad log\left(\frac{exp(x[class])}{\sum_j exp(x[j])}\right) \quad , \tag{4}$$

where $\mathbf{w}$ is the vector of class weights, with each weight $w[i] \in \mathbf{w}$ computed as the inverse of the percentage of the cardinality of the class to the total number of samples. As a result, we obtain the $L_{finetune}$ loss:

$$L_{finetune} = L_{metric} + \lambda L_{orth} + L_{CE} \quad . \tag{5}$$

4. Decision support system

This section describes the proposed decision support system, which is based on DL and XAI. Fig. 6 shows the outline of the system.

Step A: We classify the query images in an explainable way by retrieving, for each image, the training samples that are closest to it in the latent space. Then, we classify the images by majority voting on the classes of the retrieved training samples. We introduce causability by attributing the classification to the fact that the query image is visually most similar to the selected training samples[2], where similarity is enforced by metric learning training.

Step B: We integrate XAI techniques for additional support of the decision by using CAM-based methods to highlight both which regions of the images contributed to the decision and which regions are most similar to the query image. Moreover, we integrate the t-SNE dimensionality reduction technique to highlight the position of the query image with respect to the images close to it in the latent space.

4.1. Explainable classification: introducing causability

First, we remove the FC layer from the model and compute the feature vector $\mathbf{f}_i$ corresponding to the last convolutional layer for each training sample i. We also compute the feature vector $\mathbf{f}_Q$ for the query image Q in the test dataset: $\mathbf{f}_Q = \text{model}(Q)$. We normalize the feature vectors via a min–max approach, with parameters computed on the feature vectors obtained from the training samples.

Second, we analyze the distances in the latent space by computing the distance of $\mathbf{f}_Q$ from each vector in the set $\{\mathbf{f}_i\}$. We obtain the set C_Q of samples close to Q by selecting the n_{close} images with the smallest distances.

$$C_Q = \text{argmin}_i \; d(\mathbf{f}_Q, \{\mathbf{f}_i\}) \quad , \tag{6}$$

with $|C_Q| = n_{close}$.

Third, we obtain the result of the classification in an explainable way by majority voting on the labels of C_Q[4]:

$$\text{result}(C_Q) = \text{majority}\{\text{label}(C_Q)\} \quad . \tag{7}$$

[4] In our experiments, it was always possible for a clear majority of the votes to be obtained.

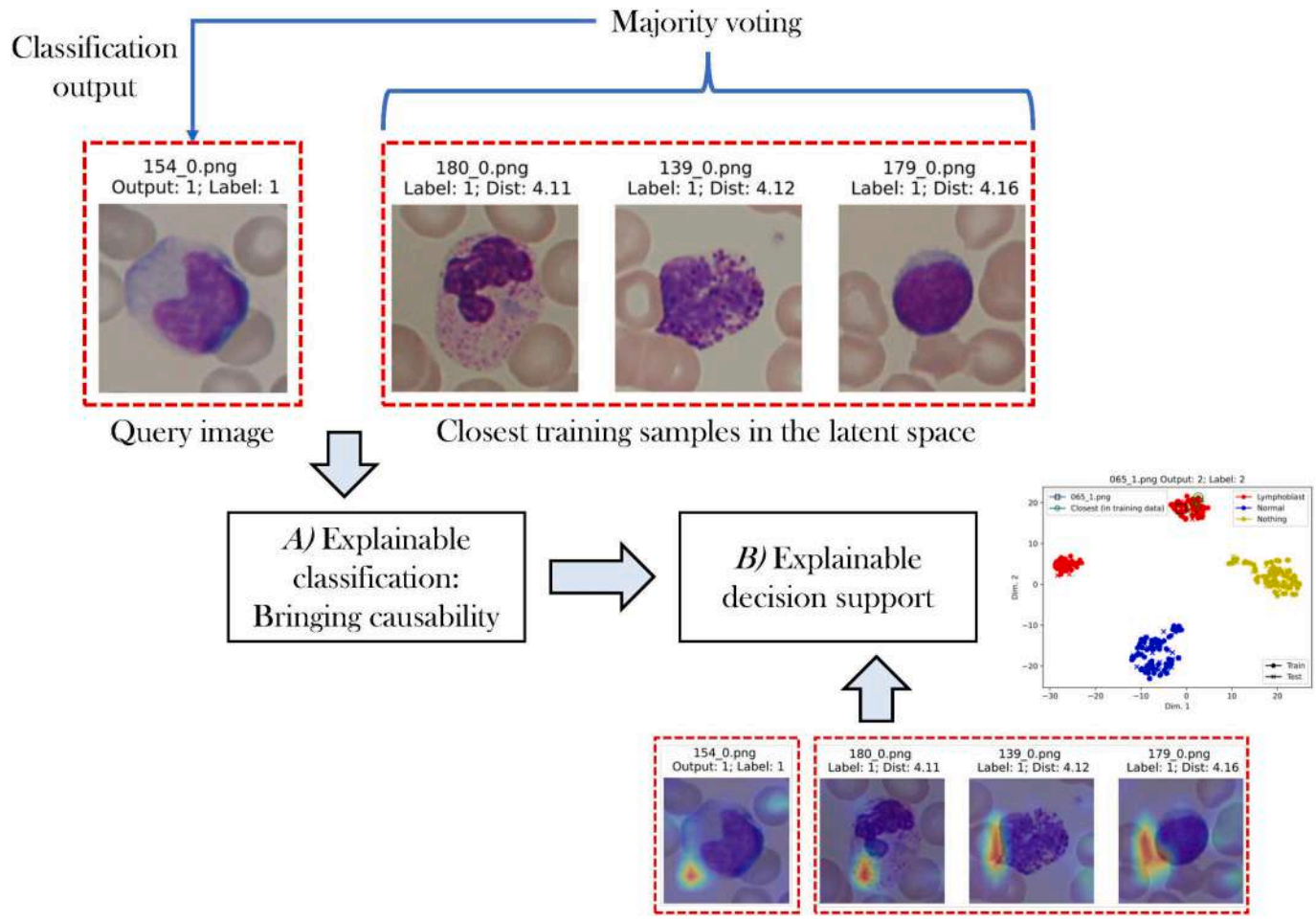

Fig. 6. Outline of the proposed decision support system. The system performs the following steps: *A)* explainable classification by majority voting on the labels of the closest training samples in the latent space. The classification introduces causability by attributing the cause of the classification to the labels of visually close images. *B)* Explainable decision support by highlighting the regions of the images that are most similar and the respective positions of the images in the latent space.

The causability of result(C_Q) lies in the fact that the classification is dependent on the labels associated with the majority of the n_{close} most similar training samples.

Fourth, we include a visual representation of the decision process by plotting the query image Q and the n_{close} images in C_Q. Fig. 7 shows an example of a query image Q and the corresponding images in C_Q used to perform majority voting. In the example, we chose an image from the "lymphoblast" class, which was correctly classified as such. In addition, for each image, we include the distances in the latent space and the corresponding labels. This representation can help in assessing the result of the classification by enabling visual examination of the images that are closer in the latent space and the corresponding distances.

4.2. Explainable decision support

First, we apply the Grad-CAM [42] method to analyze which parts of the images contributed to the decision. This method takes the trained model and a query image Q as input and outputs a heatmap of the regions of Q that contributed to the result of the classification:

$$\text{heatmap}_Q = \text{Grad-CAM}(Q, \textit{model}) \quad . \tag{8}$$

In particular, we apply this method both to the query image Q and to the set C_Q of n_{close} samples similar to Q. Fig. 8 shows a visual representation of the application of Grad-CAM to Q and C_Q, with the images in C_Q ordered by distance in the latent space and the corresponding label reported. The figure shows which parts of the image contributed to the classification of each single image. In some cases, the model focused on regions external to the WBC, providing explainable insight into whether to trust a specific prediction.

Second, we apply a method to Q and C_Q that is based on Grad-CAM but is designed to highlight which regions of a probe image are most similar to a query image [43]. The method, which we call *Metric-Grad-CAM*, takes as input the trained model, the query image Q, and a probe image P and outputs a similarity heatmap for both Q and P:

$$\left[\text{heatmap}_Q, \text{heatmap}_P\right] = \text{Metric-Grad-CAM}(Q, P, \textit{model}) \quad . \tag{9}$$

Fig. 9 shows a visual representation of the application of Metric-Grad-CAM to Q and C_Q. From the figure, it is possible to observe that the heatmaps are much more localized with respect to those in Fig. 8, showing small regions of the images that are actually similar between Q and each image in C_Q. This feature enables a much more effective and localized analysis of the images in support of the decision by focusing on small regions of the WBC that could indicate a tumoral cell.[5] In particular, the heatmaps in Fig. 9 in most cases encompass the nucleus, the cytoplasm, and the border of the cell, suggesting that it is necessary to analyze these elements to make an informed decision, in accordance with the criteria defined for analyzing lymphoblasts [44].

(a) Q and C_Q are close together, and the image Q is correctly classified (*high confidence*);

[5] We strongly discourage the use of our method for diagnostic activities without the advice of a pathologist. At this stage, our method must be considered only for image processing.

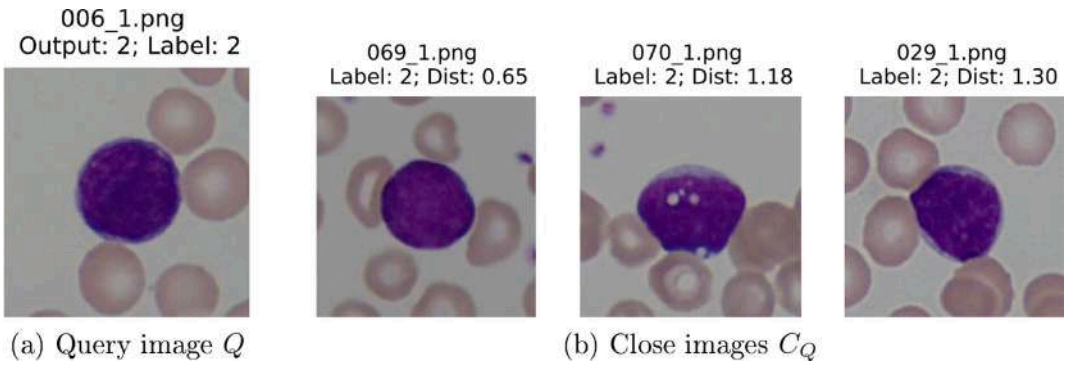

Fig. 7. Explainable classification. The proposed visual representation with the query image Q *(a)* and the n_{close} images in C_Q *(b)*. We order the images in C_Q by distance in the latent space and report the corresponding labels. This representation helps in assessing the result of the classification of Q by enabling visual examination of the images that are closer in the latent space. In the example, we chose an image from the "lymphoblast" class, which was correctly classified as such.

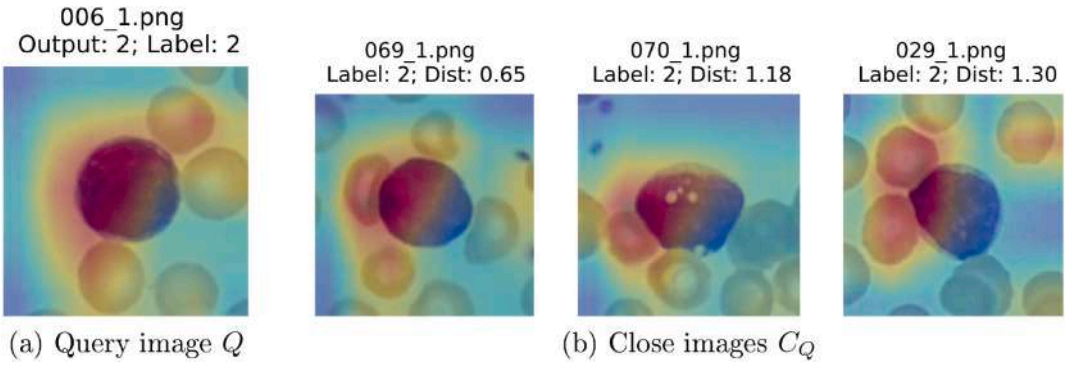

Fig. 8. Explainable decision support. The proposed visual representation is given for Grad-CAM applied to the query image Q *(a)* and to the n_{close} images in C_Q *(b)*. In addition, we order the images in C_Q by distance in the latent space and report the corresponding labels. It is possible to observe which parts of the image contributed to the classification of each single image by the trained model: in some cases, the model also focused on regions external to the WBC, providing additional insight into whether a specific classification can be trusted.

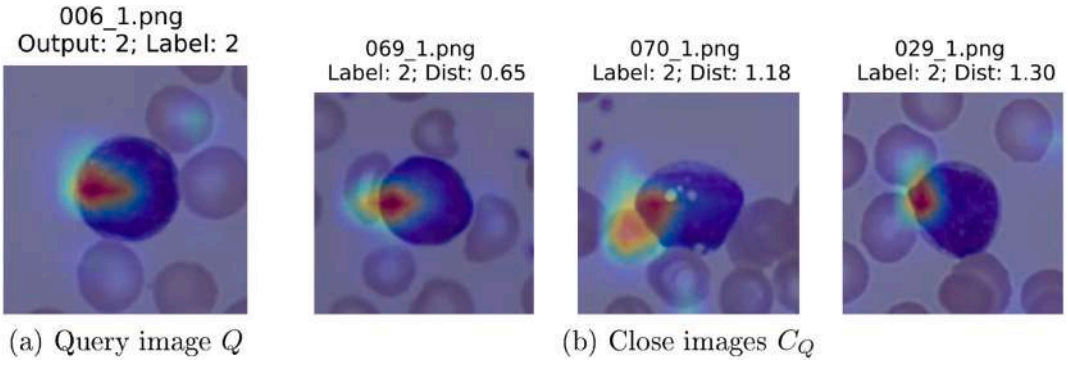

Fig. 9. Explainable decision support. The proposed visual representation is given for Metric-Grad-CAM applied to the query image Q *(a)* and to the n_{close} images in C_Q *(b)*. In addition, we order the images in C_Q by distance in the latent space and report the corresponding labels. It is possible to observe which parts of the images are similar between Q and each image in C_Q. Moreover, the heatmaps are more localized with respect to those obtained with Grad-CAM, enabling the method to focus on small regions that could indicate a tumoral cell[5].

(b) Q is significantly far from C_Q, and the image is incorrectly classified (*low confidence*).

Third, we give a visual representation of the positions of Q and C_Q in the latent space by applying the t-SNE dimensionality reduction technique [45] to both training and test data. We highlight in the t-SNE graph the positions of Q and C_Q, which can provide additional support for the decision. The graph may show that Q and C_Q are close together in the latent space, resulting in high confidence in the classification, or that they are scattered and close to the clusters of different classes, resulting in low confidence in the classification. Fig. 10 shows the proposed visual representation, plotting both training and testing samples and highlighting the query image Q and the close images in C_Q. In particular, we plot the representations for two cases: in the first case, Q and C_Q are close together in the latent space (Fig. 10a), and the image Q is correctly classified; in the second case (Fig. 10b), Q is significantly far from C_Q, and the image is incorrectly classified. The proposed visual representation can help in assigning confidence to the classification; for example, in the second case (Fig. 10b), the fact that Q and C_Q are scattered and close to the clusters of other classes will result in assigning low confidence to the decision.

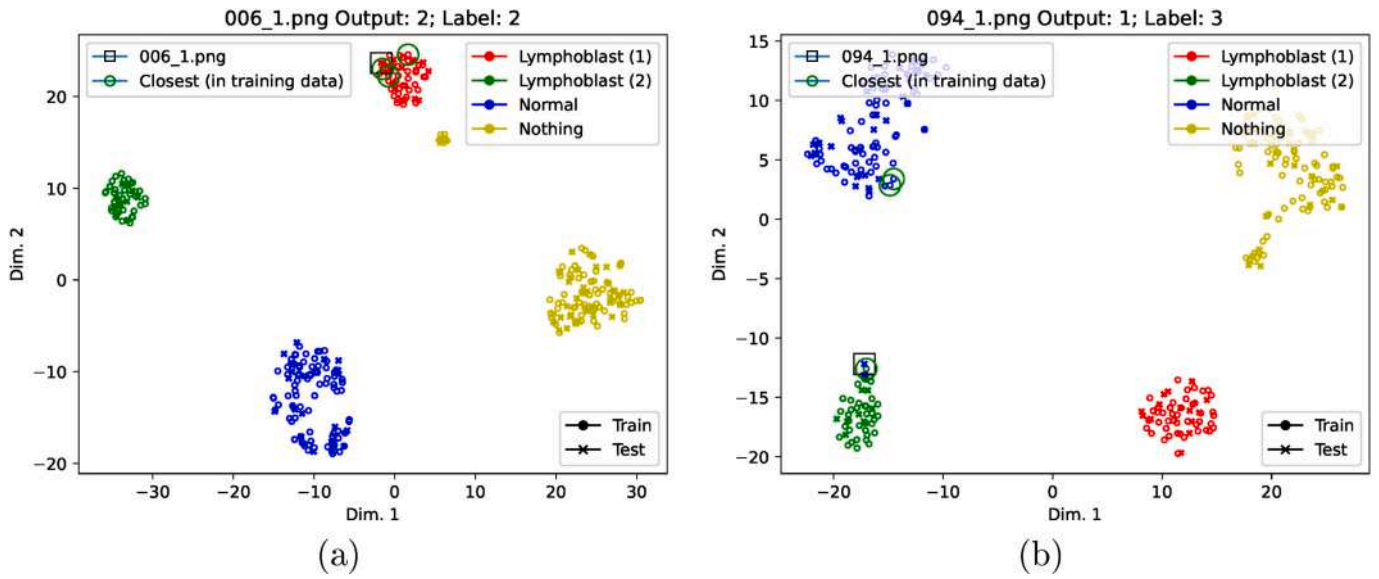

Fig. 10. Explainable decision support. The proposed visual representation is given for the t-SNE technique applied to both training and testing data. We also highlight the positions of the query image Q (*square*) and the close images in C_Q (*circles*), which can provide additional support for the decision and help assign a confidence level to the classification:

5. Experimental results

This section presents the experimental results, including the databases, model and training parameters used, and the results of the proposed methodology, both quantitative and qualitative. Finally, we present examples of the proposed decision support system.

5.1. Databases used

- *ADP:* To perform histopathological pretraining (Section 3.1), we consider the Atlas of Digital Pathology (ADP) [40],[6] which consists of 17,668 patches extracted from 100 WSIs. Each patch is individually labeled with a description of the tissues present in the patch. Each patch can contain multiple tissues; therefore, the labels are not mutually exclusive. For each patch, three different labels are available, with increasing levels of precision. We consider this database since it has patch-level labeling and is one of the few general-purpose histopathological databases; the images are of several different tissues, making it a viable option for pretraining a model for ALL detection [10,19].
- *ALL-IDB2:* This database contains 260 images of WBCs, with 130 images of "normal" WBCs and 130 images of potential lymphoblasts. Each image is then associated with a binary label (0: *normal*; 1: *lymphoblast*) [1]. The acquisition procedure used an optical microscope and a Canon PowerShot G5 camera. The acquisitions used different magnifications of the microscope, ranging from 300x to 500x, and obtained images with sizes of $H \times W = 256 \times 256$ pixels and 24-bit color depth. The images are not segmented and are already cropped and centered on the WBCs. The cropping process is relatively straightforward when analyzing images stained via the hematoxylin and eosin technique. In such images, the WBCs are highly visible, and even inexperienced operators can pinpoint the center of each cell (see Fig. 1). We enhance the database by applying the unsharpening method described in [38] and performing z score normalization, transforming the images to have a mean of 0 and standard deviation of 1. The normalization parameters are computed on the training subset. We perform class injection on the ALL database as described in Section 3.2 by first adding 130 images taken from patches that do not contain any WBCs extracted from WSIs of ALL-IDB1 via the ALL-IDB Patches approach [9]. We use 130 images since this is the number of images belonging to the class "normal" and to the class "lymphoblast" in the ALL-IDB2 database. Second, we split the images of the class "lymphoblast" into "lymphoblast (1)" and "lymphoblast (2)", on the basis of their bimodal distribution, as described in Section 3.2. As a result, we obtain a database with 4 classes: (0: *nothing*; 1: *normal*; 2: *lymphoblast (1)*; 3: *lymphoblast (2)*). The class distribution is described in Table 1. Examples of images of the corresponding classes are shown in Fig. 11.
- *ALL-IDB Patches:* This dataset contains 10,260 images of WBCs obtained by cropping patches from WSIs via the methodology described in [9]. The patches can contain red blood cells, "normal" WBCs, or lymphoblasts, and it is possible for a patch to contain multiple elements. To evaluate the proposed methodology on the dataset, we adjust it for single-output classification by considering only the patches associated with a single class. We then apply the proposed class injection method, similar to the ALL-IDB2 dataset. As a result, we obtain a database with 4 classes: 0: *nothing*; 1: *normal*; 2: *lymphoblast (1)*; 3: *lymphoblast (2)*. We enhance the database by performing z score normalization. The class distribution is described in Table 2.
- *C-NMC:* This dataset comprises 10,661 images of WBCs belonging to the classes *0*: normal and *1*: lymphoblasts. The images were captured with a Nikon DS5M camera and labeled by an expert pathologist. The images were then stain normalized to have uniform color, cropped around the cell, and segmented to show only the region of the WBC, resulting in images with size $H \times W = 450 \times 450$ pixels and 24 bit color depth [46]. We enhance the database by applying the

Table 1
Number of samples for each class in the `ALL-IDB2` database after class injection.

Class	Label	Num. of samples
0	Nothing	130
1	Normal	130
2	Lymphoblast (1)	73
3	Lymphoblast (2)	57
–	*Total*	390

[6] https://www.dsp.utoronto.ca/projects/ADP

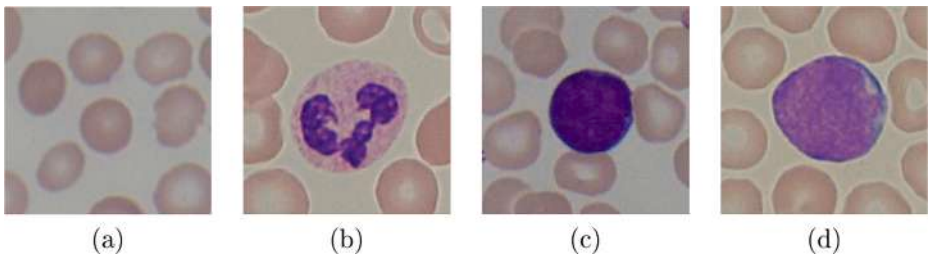

Fig. 11. Examples of images from the ALL database after class injection. *(a)* 0: *nothing*; *(b)* 1: *normal*; *(c)* 2: *lymphoblast (1)*; *(d)* 3: *lymphoblast (2)*.

Table 2
Number of samples for each class in the ALL-IDB Patches database after class injection.

Class	Label	Num. of samples
0	Nothing	6921
1	Normal	1829
2	Lymphoblast (1)	232
3	Lymphoblast (2)	336
–	*Total*	9318

unsharpening method described in [47] and performing z score normalization. The class distribution is described in Table 3.

- *PBC:* This dataset contains 17,092 images of "normal" WBCs, captured via a CellaVision DM96; they belong to 8 classes, representing different kinds of healthy WBCs: 0: *Basophil*; 1: *Eosinophil*; 2: *Erythroblast*; 3: *Immature granulocyte*; 4: *Lymphocyte*; 5: *Monocyte*; 6: *Neutrophil*; 7: *Platelet*. The images are annotated by expert pathologists and have a size of $H \times W = 360 \times 363$ pixels and 24-bit color depth [48]. We enhance the database by performing z score normalization. The class distribution is described in Table 4.

5.2. Model and training parameters

As models, we consider both CNNs and ViT, given their successful application in several fields, including histopathological image classification and ALL detection [19,40,49,50]. In particular, we consider the most used CNN variants ResNet18, ResNet34, ResNet50, and the vit_b_16.

We train the models via the multitask histopathological pretraining method described in Section 3.1, which uses the parameters and learning procedure detailed in [25]. When training vit_b_16, we consider the model pretrained on the IMAGENET-21 K database and fine tuned on IMAGENET-1 K,[7] then apply the histopathological pretraining. Before performing the metric learning fine-tuning method described in Section 3.3, we apply a warmup phase, using the ALL database, by training only the last FC layers of the model. The purpose of this warmup is to prevent gradients that are too large from initially flowing on the last FC layers since these layers are randomly initialized but are preceded by convolutional layers that are pretrained on the histopathological database [39]. In particular, in the warmup phase, we apply an SGD for 2 epochs, with a learning rate of $lr_{warmup} = 0.02$, a momentum of $= 0.9$, and a weight decay of $= 5e^{-4}$. After the warmup, we perform metric learning fine-tuning by applying an SGD for 90 epochs, with a learning rate of $lr_{finetune} = 2e^{-4}$, a momentum of $= 0.9$, and a weight decay of $= 5e^{-4}$. After every 20 epochs, we reduce the learning rate by half. We consider a deep tuning approach, enabling gradient update on all layers of the model [51].

Table 3
Number of samples for each class in the C-NMC database.

Class	Label	Num. of samples
0	Normal	3389
1	Lymphoblast	7272
–	*Total*	10,661

[7] https://github.com/lukemelas/PyTorch-Pretrained-ViT

Table 4
Number of samples for each class in the PBC database.

Class	Label	Num. of samples
0	Basophil	1218
1	Eosinophil	3117
2	Erythroblast	1551
3	Immature granulocyte	2895
4	Lymphocyte	1214
5	Monocyte	1420
6	Neutrophil	3329
7	Platelet	2348
–	*Total*	17,092

When considering the ALL-IDB Patches, C-NMC, and PBC databases, we removed the warmup phase and the orthogonal loss L_{orth}. We experimentally evaluated that such operations reduce the accuracy in those databases.

To reduce the possibility of overfitting, we perform data augmentation during training. In particular, we only consider data augmentation techniques that do not distort the image pattern, such as random rotations and random horizontal/vertical flips. To avoid distorting the image pattern, we do not consider zooming or shifting.

We test the generalizability of the model via n-fold cross-validation, with $n = 5$. In each iteration, the training subset contains approximately 3/5 of the images, whereas the validation and testing subsets contain approximately 1/5 of the images each. At the end of the training, we consider the model with the weights corresponding to the best accuracy on the validation subset. We report the final results after averaging the accuracy on the testing subset across 5 iterations.

We perform testing via the explainable classification step described in Section 4.1, which uses $n_{close} = 3$ neighbors for the ALL-IDB2 database and $n_{close} = 5$ for the others. We use the Euclidean distance function; however, other distance functions can be applied.

5.3. Results

5.3.1. Quantitative evaluation

As an error measure, we consider the classification accuracy as the percentage of samples that are correctly classified with respect to the total number of images in the testing subset of the ALL database. As additional measures, we compute the specificity and sensitivity, as well as the confusion matrix indicating the percentages of true positives, true negatives, false positives, and false negatives, according to the specifications reported in [1].

Table 5 reports the error measures using the proposed methodology on the ALL-IDB2 database compared with those of recent works in the literature. All the approaches listed in the table use histopathological

Table 5
Accuracy results on the ALL-IDB2 database.

Ref.	Model	Accuracy* (%) ($Mean_{Std}$)	Sensitivity*	Specificity*
[38]	ResNet18	$96.00_{1.13}$	$94.76_{2.82}$	$97.23_{2.02}$
[19]	$HistoTNet_{ResNet18}$	$97.23_{1.15}$	$97.54_{2.02}$	$96.92_{2.43}$
	$HistoTNet_{ResNet34}$	$98.31_{1.91}$	$98.15_{3.34}$	$98.46_{1.09}$
[32]	$ALLNet_{ResNet18}$	$97.85_{0.58}$	$98.46_{1.54}$	$97.23_{2.01}$
	$ALLNet_{ResNet34}$	$98.46_{0.84}$	$97.85_{1.75}$	$99.08_{1.38}$
[9]	$OrthoALLNet_{ResNet18}$	$98.46_{0.97}$	$99.08_{0.84}$	$97.85_{2.33}$
	$OrthoALLNet_{ResNet34}$	$98.62_{0.90}$	$97.85_{2.06}$	$\mathbf{99.38_{0.84}}$
	ResNet18	$\mathbf{99.74_{0.51}}$	$\mathbf{100.00_{0.00}}$	$99.26_{1.66}$
	ResNet34	$99.23_{0.63}$	$98.57_{1.96}$	$98.95_{2.35}$
	ResNet50	$99.74_{0.51}$	$100.00_{0.00}$	$99.26_{1.66}$
	vit_b_16	$98.21_{1.46}$	$94.79_{4.03}$	$100.00_{0.00}$

Notes: * We computed the accuracy, sensitivity, and specificity for our approach after processing the confusion matrix combining class 0: *nothing* with class 1: *normal* and combining class 2: *lymphoblast (1)* with class 3: *lymphoblast (2)*.

pretraining and the same training hyperparameters. The table shows that our approach achieves superior accuracy with respect to recent works in the literature. In particular, ResNet18 and ResNet50 achieve the best result, with a classification accuracy of 99.74%. With respect to the methods in the literature, our method has the advantages of increased accuracy and an added decision support system, which increases the causability and explainability of the decisions. Table 6 also presents the corresponding confusion matrix, which shows that the class injection does not negatively affect the accuracy, since the only classification errors are between the "normal" WBCs and the lymphoblasts. In fact, no image of class 0: *nothing* was misclassified, and no image of class 2: *lymphoblast (1)* was misclassified as class 3: *lymphoblast (2)* or vice versa.

Table 7, Table 8, and Table 9 present the results of the proposed methodology on the ALL-IDB Patches, C-NMC, and PBC databases, respectively. In the tables, we compare the results of our methodology with those of the approaches presented in [9,10,52], which exhibit state-of-the-art accuracy on the databases.

Table 7 and Table 8 show that our approach achieves a greater accuracy with respect to recent works in the literature also when applied on the ALL-IDB Patches and C-NMC databases. In particular, for ALL-IDB Patches the ResNet34 achieves the best result, with a classification accuracy of 99.20%, while for C-NMC the vit_b_16 achieves the best result with 97.32% accuracy. It is worth noting that for ALL-IDB2 and ALL-IDB Patches we applied the class injection step, which enabled the models trained on these databases to more accurately distinguish between classes. Table 9 shows that the proposed methodology achieves similar results to those of the methods from the literature, with a slight decrease in accuracy caused by the use of a simpler classifier (majority voting). In particular, on the PBC database we obtain a best result of 99.00%, only 0.91% less than the current literature best (99.91%). However, the proposed method is more explainable than the methods from current literature. To increase the accuracy, it would also be possible to combine the proposed approach with different architectures and training algorithms, provided that a regularized latent space is obtained.

Table 6
Average confusion matrix of the results obtained using the ResNet18 configuration on the ALL-IDB2 database.

		Predicted			
		0: *nothing*	1: *normal*	2: *lymph. (1)*	3: *lymph. (2)*
True	0: *nothing*	33.33%	0%	0%	0%
	1: *normal*	0%	33.07%	0%	0.25%
	2: *lymph. (1)*	0%	0%	18.71%	0%
	3: *lymph. (2)*	0%	0%	0%	14.64%

Table 7
Accuracy results on the ALL-IDB Patches database.

Ref.	Model	Accuracy* (%) ($Mean_{Std}$)	Sensitivity*	Specificity*
[9]	$OrthoALLNet_{ResNet18}$	$95.91_{0.81}$	–	–
	$OrthoALLNet_{ResNet34}$	$96.06_{0.78}$	–	–
	ResNet18	$99.14_{0.18}$	$95.92_{1.26}$	$99.35_{0.14}$
	ResNet34	$\mathbf{99.20_{0.18}}$	$\mathbf{96.01_{2.27}}$	$99.39_{0.15}$
	ResNet50	$98.99_{0.25}$	$94.76_{1.38}$	$99.27_{0.26}$
	vit_b_16	$99.13_{0.14}$	$94.39_{1.32}$	$\mathbf{99.44_{0.23}}$

Notes: * We computed the accuracy, sensitivity, and specificity for our approach after processing the confusion matrix combining class 0: *nothing* with class 1: *normal* and combining class 2: *lymphoblast (1)* with class 3: *lymphoblast (2)*.

Table 8
Accuracy results on the C-NMC database.

Ref.	Model	Accuracy (%) ($Mean_{Std}$)	Sensitivity	Specificity
[10]	$ResNet18_{ADP,3,CNMC}$	$95.99_{0.16}$	$90.94_{1.33}$	$98.35_{0.54}$
	$ViT_{ImageNet,ADP,1,CNMC}$	$97.10_{0.38}$	$94.00_{1.09}$	$\mathbf{98.54_{0.25}}$
	ResNet18	$95.86_{0.57}$	$97.81_{0.40}$	$91.71_{1.56}$
	ResNet34	$95.66_{0.51}$	$97.68_{0.52}$	$91.31_{1.09}$
	ResNet50	$95.01_{0.41}$	$97.26_{0.17}$	$90.23_{0.79}$
	vit_b_16	$\mathbf{97.32_{0.22}}$	$\mathbf{99.01_{0.13}}$	$93.67_{0.88}$

Table 9
Accuracy results on the PBC database.

Ref.	Model	Accuracy (%) ($Mean_{Std}$)
[52]	Custom CNN	99.91
	ResNet18	$98.95_{0.13}$
	ResNet34	$98.99_{0.25}$
	ResNet50	$99.00_{0.10}$
	vit_b_16	$98.69_{0.18}$

5.3.2. Qualitative evaluation

To perform the qualitative evaluation, we consider the results obtained on ALL-IDB2 using the ResNet18 CNN, which obtained the highest classification accuracy (see Table 5). In particular, we apply the t-SNE dimensionality reduction technique in the "standard configuration"; it is applied to the testing subset of the data, considering the feature space obtained after the last convolutional layer of the CNN (the same feature space used for metric learning and distance computation). This analysis is routinely performed in several papers to show the effectiveness of the training [10]. Fig. 12 shows the results on the ALL-IDB2 database, indicating that the four classes are effectively separated and that class 2: *lymphoblast (1)* and class 3: *lymphoblast (2)* are separated, supporting the method of separating the classes during the class injection step (Section 3.2).

5.3.3. Ablation study

To evaluate the effects of class injection and metric learning, we analyze them separately by computing the accuracy with and without them on the ALL-IDB2 database. We use ResNet18 since it provided the best accuracy (see Table 5). Table 10 shows the obtained results. The table shows that the combined use of metric learning and class injection allows us to obtain the best results.[8]

[8] Because the accuracy of the proposed methodology is almost saturated, we did not perform a sensitivity analysis for the training hyperparameters.

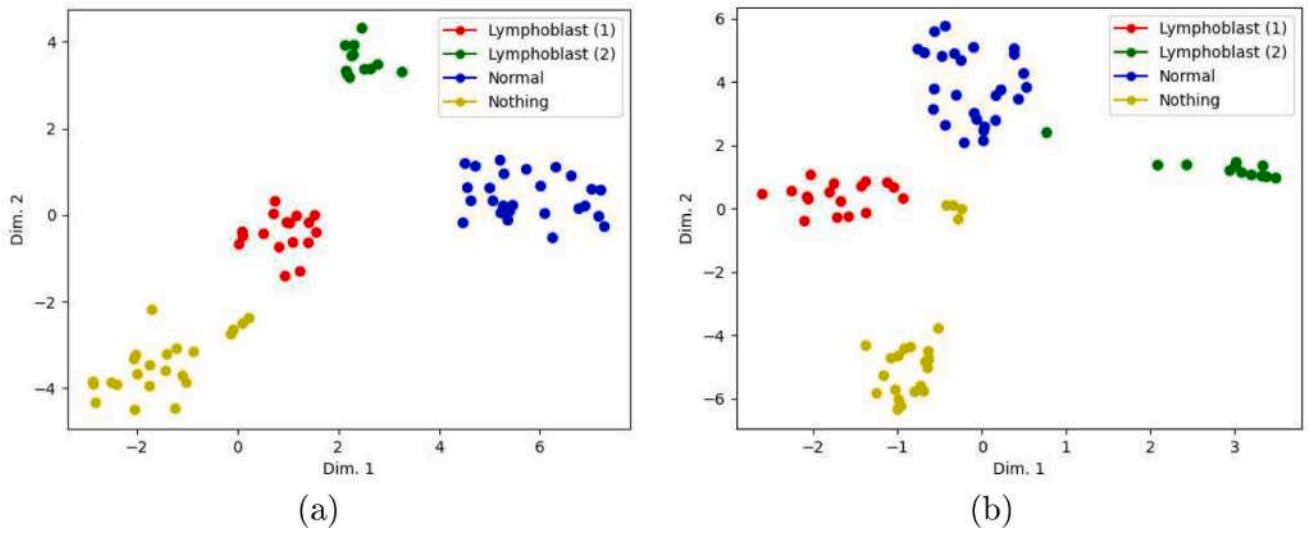

Fig. 12. Results of the t-SNE algorithm in the "standard configuration", applied to the testing subset of the ALL-IDB2 database and considering the feature space obtained after the last convolutional layer of the CNN: *(a)* ResNet18; *(b)* ResNet34. In both cases, it is seen that the four classes are effectively separated and that class 2: *lymphoblast (1)* and class 3: *lymphoblast (2)* are distinct.

Table 10
Ablation study of the proposed methodology on the ALL-IDB2 database.

Model	Metric learning	Class injection	Accuracy (%) (Mean$_{Std}$)
ResNet18			$99.49_{0.63}$
		✓	$99.49_{0.63}$
	✓		$98.97_{0.96}$
	✓	✓	$\mathbf{99.74_{0.51}}$

5.3.4. Running times

The hardware used in our experiments consists of a laptop computer equipped with a 12th Gen Intel(R) Core(TM) i7-12800H @2.40 GHz processor, 64 GB RAM and an Nvidia RTX A1000 GPU. The training process for the ResNet18 took between 2 m 45 s and 3 m for 90 epochs and 234 images. For comparison, the training process took approximately 2 m for a method based on a plain ResNet [19]. The inference times for both methods were approximately 1 s for 78 images. We did not consider visualization since *i)* saving and visualizing plots also depends on hard drive speed and *ii)* to reduce the inference time, it would be possible to request only the visualization of a specific sample. Moreover, to evaluate the scalability of our approach in the testing phase, we tripled the size of the dataset using a dummy replication scheme, with no significant difference in inference time. Finally, we believe that further speed improvements could be achieved by considering techniques for optimizing DL models [53,54].

5.4. Decision support system

We evaluate the proposed decision support system by presenting examples of the visual representations associated with a classification. In each representation, we include one example of a correct classification and one example of an incorrect classification.

First, we present the visual representation for the explainable classification, as described in Section 4.1. The representation includes the query image Q, the images C_Q close in the feature space used to perform the classification, and the distances in the feature space. Fig. 13 is the visual representation, which shows that in a correct classification, the images in C_Q are significantly more similar to those in Q than for an incorrect classification. Moreover, in the case of a correct classification, the average distance between Q and C_Q is < 1, which is significantly lower than that of an incorrect classification, where the average distance is > 4.

Second, we present the visual representation with Metric-Grad-CAM, as described in Section 4.2. Fig. 14 shows the proposed explainable decision support method based on Metric-Grad-CAM. The figure shows that in the correct classification, Metric-Grad-CAM highlights regions of the images in C_Q within the WBCs that are similar to those in the query image Q, in contrast to the case of incorrect classification, where Metric-Grad-CAM highlights regions that are not included within the WBC and therefore are probably not significant[5].

Third, we present the visual representation with the t-SNE dimensionality reduction technique applied to both the training and testing data, highlighting in the t-SNE graph the positions of Q and C_Q, as described in Section 4.2. Fig. 15 shows the proposed explainable decision support based on t-SNE, showing that for the correct classification, Q and C_Q are much closer than they are in the incorrect classification case.

6. Conclusions

In this paper, we propose the first decision support system for ALL detection; it is based on deep learning and explainable artificial intelligence, with the purpose of providing an informed decision by introducing causability into the classification result. Our approach first trains a model via a metric learning approach, which organizes the latent spaces and brings samples belonging to the same class closer to each other and farther from samples of other classes. Second, the approach performs explainable classification of each white blood cell as either 0: *normal* or 1: *lymphoblast* by majority voting of the closest samples in the latent space. *Colloquially speaking, our decision support system answers the classification question for an unknown sample by saying, "I classify this sample with the label 'x' because it is most similar to these images that also have the label 'x' "*. The method then uses integrated XAI techniques to provide an informed decision by considering CAM-based techniques that highlight which parts of the images are most similar to the query image and a t-SNE-based dimensionality reduction method that shows the distances between the closest samples with respect to the query image and the corresponding position.

In addition to proposing a decision support system, the advantages of our methodology reside also in the class injection, which enabled the DL models to have a larger output space to embed the results of the classification. Combined with metric learning, the result is a more regularized latent space, which can be leveraged by a simple majority voting

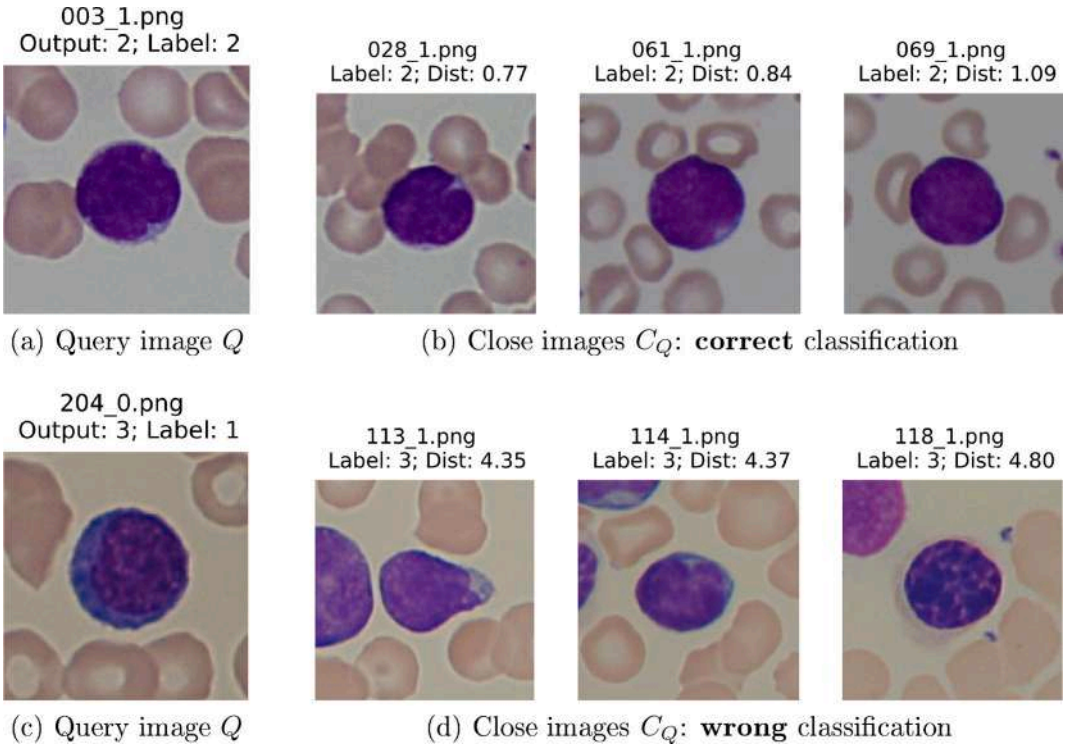

Fig. 13. Decision support system on the ALL-IDB2 database: explainable classification. Examples of the visual representation of the decision process for a correct classification (*a:* query image Q; *b:* close images C_Q) and an incorrect classification (*c:* query image Q; *d:* close images C_Q). In the correct classification, the selected close images *(b)* are very similar to the query image *(a)*. Moreover, the average distance in the feature space is < 1. On the other hand, in the incorrect classification, the close images *(d)* are somewhat dissimilar to the query image *(c)*. Moreover, the average distance in the feature space is > 4, which is significantly greater than that of correct classification.

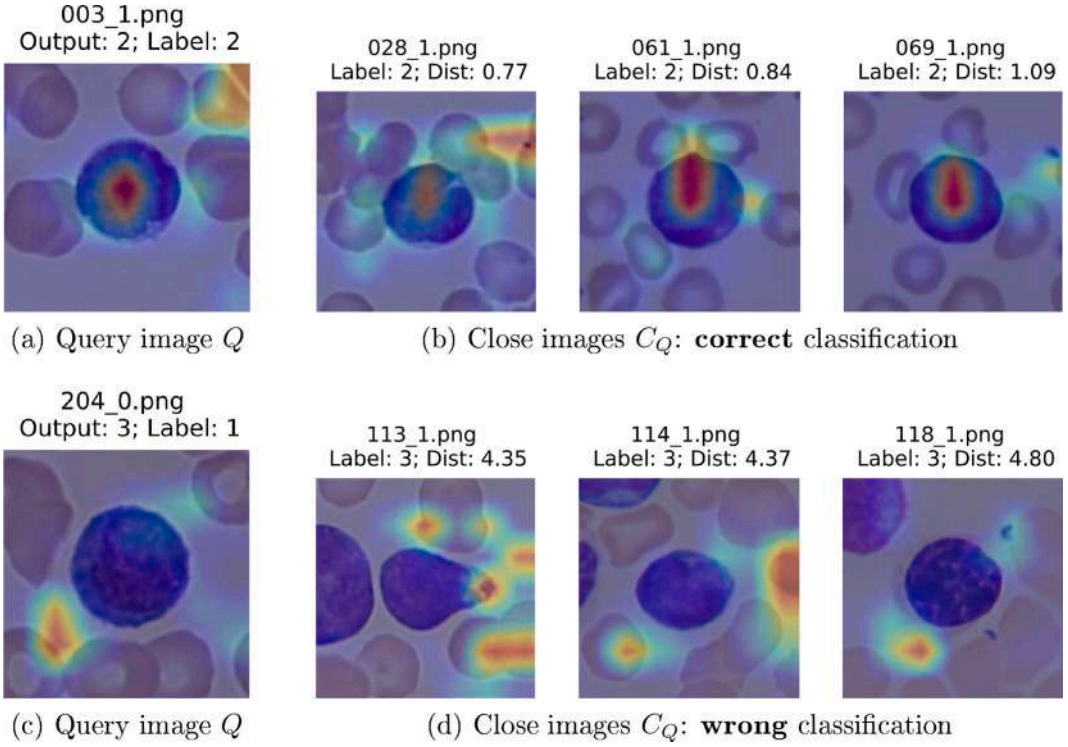

Fig. 14. Decision support system on the ALL-IDB2 database: explainable decision support. Examples of the visual representation using Metric-Grad-CAM for a correct classification (*a:* query image Q, *b:* close images C_Q) and an incorrect classification (*c:* query image Q, *d:* close images C_Q). For the correct classification, in the selected close images *(b)*, Metric-Grad-CAM highlights regions that are within the WBCs and are similar to those in the query image *(a)*. On the other hand, for the incorrect classification, Metric-Grad-CAM highlights regions that are not included within the WBC and therefore are probably not significant.

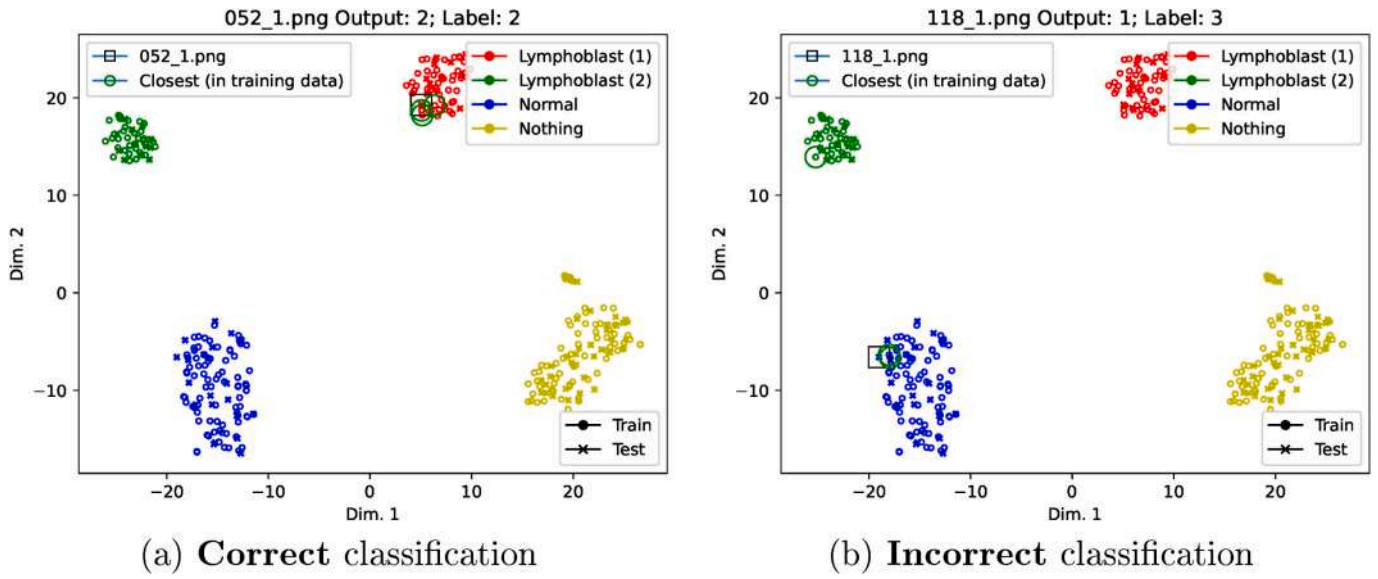

Fig. 15. Decision support system on the ALL-IDB2 database: explainable decision support. Examples of the proposed visual representation are given based on the t-SNE algorithm for correct classification *(a)* and incorrect classification *(b)*. The representation includes both training and testing data and highlights the positions of the query image Q (*square*) and the close images C_Q (*circles*) in the latent space. In the correct classification case, Q and C_Q are much closer together than in the incorrect classification case.

classifier providing an explainable decision. The results on several publicly available ALL databases demonstrate classification accuracy that is similar to or greater than that of state-of-the-art methods, at the same time introducing causability into classification while providing tools to assess the confidence of the decision. Future works will investigate the feasibility of a class injection step for other databases, as well as the transferability of the learned knowledge between different datasets.

CRediT authorship contribution statement

Angelo Genovese: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Vincenzo Piuri:** Writing – review & editing, Project administration, Funding acquisition. **Fabio Scotti:** Writing – review & editing, Writing – original draft, Resources, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported in part by the EC under the EdgeAI project (101097300) and by the Italian MUR under the SERICS project (PE00000014) of the NRRP MUR program funded by the EU-NGEU. We also thank the NVIDIA Corporation for the donated GPU. The views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union or the Italian MUR. Neither the European Union nor the Italian MUR can be held responsible for them.

References

[1] R. Donida Labati, V. Piuri, 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, M.G. Oghli, Recognition of acute lymphoblastic leukemia cells in microscopic images using k-means clustering and support vector machine classifier, J. Med. Sign. Sens. 5 (1) (Jan. 2015).
[3] H.T. Salah, I.N. Muhsen, M.E. Salama, T. Owaidah, S.K. Hashmi, Machine learning applications in the diagnosis of leukemia: current trends and future directions, Int. J. Lab. Hematol. 41 (6) (Dec. 2019).
[4] S. Kulkarni, N. Seneviratne, M.S. Baig, A.H.A. Khan, Artificial intelligence in medicine: where are we now? Acad. Radiol. 27 (1) (Jan. 2020) special Issue: Artificial Intelligence.
[5] H. Shin, H.R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, R. M. Summers, Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning, IEEE Trans. Med. Imag. 35 (5) (2016) 1285–1298.
[6] L.D. Biasi, A.A. Citarella, M. Risi, G. Tortora, A cloud approach for melanoma detection based on deep learning networks, IEEE J. Biomed. Health Inform. 26 (3) (2022) 962–972.
[7] M.S. Hosseini, B.E. Bejnordi, V.Q.-H. Trinh, L. Chan, D. Hasan, X. Li, S. Yang, T. Kim, H. Zhang, T. Wu, K. Chinniah, S. Maghsoudlou, R. Zhang, J. Zhu, S. Khaki, A. Buin, F. Chaji, A. Salehi, B.N. Nguyen, D. Samaras, K.N. Plataniotis, Computational pathology: a survey review and the way forward, J. Pathol. Inform. 15 (2024) 1–72.
[8] M. Zolfaghari, H. Sajedi, A survey on automated detection and classification of acute leukemia and WBCs in microscopic blood cells, Multimed. Tools Appl. 81 (2022) 6723–6753.
[9] A. Genovese, V. Piuri, F. Scotti, ALL-IDB patches: Whole slide imaging for acute lymphoblastic leukemia detection using deep learning, in: Proc. of ICASSPW, 2023.
[10] A. Genovese, V. Piuri, K.N. Plataniotis, F. Scotti, DL4ALL: multi-task cross-dataset transfer learning for acute lymphoblastic leukemia detection, IEEE Access 11 (2023) 65222–65237.
[11] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, H. Müller, Causability and explainability of artificial intelligence in medicine, WIREs Data Min. Knowl. 9 (4) (Jul. 2019).
[12] W. Saeed, C. Omlin, Explainable AI (XAI): a systematic meta-survey of current challenges and future opportunities, Knowl.-Based Syst. 263 (2023) 110273.
[13] Q. Teng, Z. Liu, Y. Song, K. Han, Y. Lu, A survey on the interpretability of deep learning in medical diagnosis, Multimedia Systems 28 (6) (2022) 2335–2355.
[14] S. Tsutsui, W. Pang, B. Wen, WBCAtt: A white blood cell dataset annotated with detailed morphological attributes, in: Proc. of NIPS, 2023.
[15] J. van der Laak, G. Litjens, F. Ciompi, Deep learning in histopathology: the path to the clinic, Nat. Med. 27 (2021) 775–784.
[16] S. Mishra, B. Majhi, P.K. Sa, L. Sharma, Gray level co-occurrence matrix and random forest based acute lymphoblastic leukemia detection, Biomed. Signal Process. 33 (Mar. 2017).
[17] J. Rawat, A. Singh, H.S. Bhadauria, J. Virmani, J.S. Devgun, Classification of acute lymphoblastic leukaemia using hybrid hierarchical classifiers, Multimed. Tools Appl. 76 (18) (Sep. 2017).

[18] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[19] A. Genovese, M.S. Hosseini, V. Piuri, K.N. Plataniotis, F. Scotti, Histopathological transfer learning for acute lymphoblastic leukemia detection, in: Proc. of CIVEMSA, 2021.
[20] B. Masoudi, VKCS: A pre-trained deep network with attention mechanism to diagnose acute lymphoblastic leukemia, Multimed. Tools Appl. 82 (Nov 2022) 18967–18983.
[21] A. Loddo, L. Putzu, On the effectiveness of leukocytes classification methods in a real application scenario, AI 2 (3) (2021) 394–412.
[22] S. Shafique, S. Tehsin, Acute lymphoblastic leukemia detection and classification of its subtypes using pretrained deep convolutional neural networks, Technol. Cancer Res. Trans. 17 (Jan. 2018).
[23] A. Rehman, N. Abbas, T. Saba, S.I.U. Rahman, Z. Mehmood, H. Kolivand, Classification of acute lymphoblastic leukemia using deep learning, Microsc. Res. Tech. 81 (11) (Nov. 2018).
[24] N. Tajbakhsh, J.Y. Shin, S.R. Gurudu, R.T. Hurst, C.B. Kendall, M.B. Gotway, J. Liang, Convolutional neural networks for medical image analysis: full training or fine tuning? IEEE Trans. Med. Imag. 35 (5) (2016) 1299–1312.
[25] A. Genovese, V. Piuri, F. Scotti, ALL-IDB Patches: Whole slide imaging for acute lymphoblastic leukemia detection using deep learning, in: Proc. of ICASSP, 2023, pp. 1–5.
[26] A. Talaat, P. Kollmannsberger, A. Ewees, Efficient classification of white blood cell leukemia with improved swarm optimization of deep features, Sci. Rep. 10 (2536) (Feb. 2020).
[27] D.S. Depto, M.M. Rizvee, A. Rahman, H. Zunair, M.S. Rahman, M. Mahdy, Quantifying imbalanced classification methods for leukemia detection, Comput. Biol. Med. 152 (2023) 106372.
[28] P. Mathur, M. Piplani, R. Sawhney, A. Jindal, R.R. Shah, Mixup multi-attention multi-tasking model for early-stage leukemia identification, in: Proc. of ICASSP, 2020.
[29] M. Kaur, A.A. AlZubi, A. Jain, D. Singh, V. Yadav, A. Alkhayyat, DSCNet: deep skip connections-based dense network for all diagnosis using peripheral blood smear images, Diagnostics 13 (17) (2023).
[30] M.E. Billah, F. Javed, Bayesian convolutional neural network-based models for diagnosis of blood cancer, Appl. Artif. Intell. 1–22 (2021).
[31] A. Kumar, J. Rawat, I. Kumar, M. Rashid, K.U. Singh, Y.D. Al-Otaibi, U. Tariq, Computer-aided deep learning model for identification of lymphoblast cell using microscopic leukocyte images, Expert. Syst. 39 (4) (2021) 1–13, e12894.
[32] A. Genovese, ALLNet: Acute lymphoblastic leukemia detection using lightweight convolutional networks, in: Proc. of CIVEMSA, 2022.
[33] P.K. Das, B. Nayak, S. Meher, A lightweight deep learning system for automatic detection of blood cancer, Measurement 191 (2022) 110762.
[34] S. Dhalla, A. Mittal, S. Gupta, LeukoCapsNet: a resource-efficient modified capsnet model to identify leukemia from blood smear images, Neural Comput. & Applic. 36 (2023) 2507–2524.
[35] A. Askari-Farsangi, A. Sharifi-Zarchi, M.H. Rohban, Novel pipeline for diagnosing acute lymphoblastic leukemia sensitive to related biomarkers, 2023 arXiv: 2307.04014.
[36] P. Manescu, P. Narayanan, C. Bendkowski, M. Elmi, R. Claveau, V. Pawar, B. J. Brown, M. Shaw, A. Rao, D. Fernandez-Reyes, Detection of acute promyelocytic leukemia in peripheral blood and bone marrow with annotation-free deep learning, Sci. Rep. 13 (2562) (2023).
[37] R. Duggal, A. Gupta, R. Gupta, P. Mallick, SD-layer: Stain deconvolutional layer for CNNs in medical microscopic imaging, in: Proc. of MICCAI, 2017.
[38] A. Genovese, M.S. Hosseini, V. Piuri, K.N. Plataniotis, F. Scotti, Acute lymphoblastic leukemia detection based on adaptive unsharpening and deep learning, in: Proc. of ICASSP, 2021.
[39] R. Mormont, P. Geurts, R. Marée, Multi-task pre-training of deep neural networks for digital pathology, IEEE J. Biomed. Health Inform. 25 (2) (2021) 412–421.
[40] M.S. Hosseini, L. Chan, G. Tse, M. Tang, J. Deng, S. Norouzi, C. Rowsell, K. N. Plataniotis, S. Damaskinos, Atlas of digital pathology: A generalized hierarchical histological tissue type-annotated database for deep learning, in: Proc. of CVPR, 2019.
[41] J. Wang, Y. Chen, R. Chakraborty, S.X. Yu, Orthogonal convolutional neural networks, in: Proc. of CVPR, 2020.
[42] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: Visual explanations from deep networks via gradient-based localization, in: Proc. of ICCV, 2017.
[43] S. Zhu, T. Yang, C. Chen, Visual explanation for deep metric learning, IEEE Trans. Image Process. 30 (2021) 7593–7607.
[44] F. Scotti, Automatic morphological analysis for acute leukemia identification in peripheral blood microscope images, in: Proc. of CIMSA, 2005.
[45] L. van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (86) (2008) 2579–2605.
[46] R. Gupta, S. Gehlot, A. Gupta, C-NMC: B-lineage acute lymphoblastic leukaemia: a blood cancer dataset, Med. Eng. Phys. 103 (2022) 103793.
[47] M.S. Hosseini, K.N. Plataniotis, Convolutional deblurring for natural imaging, IEEE Trans. Image Process. 29 (2020) 250–264.
[48] A. Acevedo, A. Merino, S. Alférez, L. Ángel Molina, J. Rodellar Boldú, A dataset of microscopic peripheral blood cell images for development of automatic recognition systems, Data Brief 30 (2020) 105474.
[49] M.S. Hosseini, L. Chan, W. Huang, Y. Wang, D. Hasan, C. Rowsell, S. Damaskinos, K.N. Plataniotis, On transferability of histological tissue labels in computational pathology, in: Proc. of ECCV, 2020.
[50] F. Shamshad, S. Khan, S.W. Zamir, M.H. Khan, M. Hayat, F.S. Khan, H. Fu, Transformers in medical imaging: a survey, Med. Image Anal. 88 (2023) 102802.
[51] R. Zhang, J. Zhu, S. Yang, M.S. Hosseini, A. Genovese, L. Chen, C. Rowsell, S. Damaskinos, S. Varma, K.N. Plataniotis, HistoKT: Cross knowledge transfer in computational pathology, in: Proc. of ICASSP, 2022.
[52] R. Asghar, S. Kumar, P. Hynds, Automatic classification of 10 blood cell subtypes using transfer learning via pre-trained convolutional neural networks, Inform. Med. Unlock. 49 (2024) 101542.
[53] Whole Slide Image Analysis in Real Time with MONAI and RAPIDS, URL, https://developer.nvidia.com/blog/whole-slide-image-analysis-in-real-time-with-monai-and-rapids/, 2024.
[54] OpenVINO, URL, https://docs.openvino.ai/2024/index.html#, 2024.