electronics

Article

Blood Cell Attribute Classification Algorithm Based on Partial Label Learning

Junxin Feng, Qianhang Guo, Shiling Luo, Letao Chen and Qiongxiong Ma *

Guangdong Provincial Key Laboratory of Nanophotonic Functional Materials and Devices, School of Information and Optoelectronic Science and Engineering, South China Normal University, Guangzhou 510006, China; 20213231036@m.scnu.edu.cn (J.F.); guoqianhang@m.scnu.edu.cn (Q.G.); 20213232093@m.scnu.edu.cn (S.L.); 20213231011@m.scnu.edu.cn (L.C.)

* Correspondence: maqx@m.scnu.edu.cn

Abstract: Hematological morphology examinations, essential for diagnosing blood disorders, increasingly utilize deep learning. Blood cell classification, determined by combinations of cell attributes, is complicated by the complex relationships and subtle differences among the attributes, resulting in significant time and cost penalties. This study introduces the Partial Label Learning for Blood Cell Classification (P4BC) strategy, a method that trains neural networks using the blood cell attribute labeling data of weak annotations. Using morphological knowledge, we predefined candidate label sets for the blood cell attributes to blend this knowledge with deep learning. This improves the model's prediction accuracy and interpretability in classifying attributes. This method effectively combines morphological knowledge with deep learning, an approach we refer to as knowledge alignment. It results in an 8.66% increase in attribute recognition accuracy and a 1.09% improvement in matching predictions to the candidate label sets, compared to the original method. These results confirm our method's ability to grasp the characteristic information of blood cell attributes, enhancing the model interpretability and achieving knowledge alignment between hematological morphology and deep learning. Our algorithm ensures attribute classification accuracy and shows excellent cell category classification, highlighting its wide application potential and practical value in blood cell category classification.

Keywords: deep learning; blood cell image classification; partial label learning; interpretability; knowledge alignment

check for updates

Citation: Feng, J.; Guo, Q.; Luo, S.; Chen, L.; Ma, Q. Blood Cell Attribute Classification Algorithm Based on Partial Label Learning. *Electronics* **2024**, *13*, 1698. https://doi.org/10.3390/electronics13091698

Academic Editor: Maria Filomena Santarelli

Received: 2 April 2024
Revised: 20 April 2024
Accepted: 24 April 2024
Published: 27 April 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Currently, hematological morphology examination is one of the most commonly used methods for diagnosing blood diseases [1], and the application of deep learning for classification in hematological morphology has become mainstream [2]. It plays a crucial role in diagnosing diseases such as leukemia and nutritional anemia [3,4].

In recent years, with the improvement of hardware performance and advancements in deep learning technologies, numerous sophisticated deep learning models have emerged [5–9], particularly excelling in the medical field [10,11]. Undeniably, deep learning has gained a distinct advantage over traditional machine learning methods in blood cell image classification [12], leading many scholars to apply deep learning in the processing of blood cell images [13–15]. Liao et al. [16] developed a convolutional neural network based on ultrasound RF signals for the extraction of red blood cell classification features. This method not only reduced the network parameters but also improved the classification accuracy, despite not quantitatively assessing the cells. Kishore et al. [17] optimized a convolutional neural network using the particle swarm optimization algorithm for the quantitative classification assessment of white blood cells, effectively reducing the misclassification rate of white blood cells. However, this method is primarily suited for handling multiple overlapping

cells and cell clusters. Khan et al. [18] combined a convolutional neural network with a dual attention network for the efficient detection and classification of white blood cells, adapting to various cell distribution scenarios. Machine learning and radiomics methods can extract and process the features of various blood cell attributes. Wu et al. [19] introduced a method to extract the radiomic features from blood cell images. This method uses computational algorithms to convert image structures into quantitative data.

Deep learning technology has undergone wide application and significant achievements in blood cell classification within the medical field. However, critical issues remain. The authenticity of the blood cell results predicted by models lacks clear verification. This lack of verification can lead to misdiagnoses and serious medical accidents [20]. Additionally, relying only on precisely annotated cell category data is not sufficient to develop deep learning models that excel in cell attribute classification. Each blood cell image contains rich attribute information but does not become accurately annotated. This situation often causes the model's focus to scatter. It makes it difficult to effectively recognize and capture the subtle differences between attributes. Therefore, developing such models requires a large amount of precisely annotated attribute data. Considering that each blood cell image requires annotations for multiple attributes, the cost of data annotation significantly increases, posing a major challenge [21]. Hence, there is an urgent need for a blood cell attribute classification method that requires lower demands for attribute data annotation. The classification results should be consistent with the hematological morphology and interpretable. The Concept Bottleneck Model (CBM) [22] is an interpretable method. It labels tasks with human anatomy concepts related to predictions. Then, it predicts the final label based on the concept labels.

CBM moderates the classification task at an intermediate level to improve credibility. This allows for human intervention to correct wrong concept predictions, enhancing the model performance. Zarlenga et al. [23] introduced Concept Embedding Models. These models focus on learning interpretable high-dimensional concept representations. They achieved a balance between accuracy and interpretability. Yuksekgonul et al. [24] proposed Post-hoc Concept Bottleneck Models (PCBM). PCBMs can transform any neural network into a PCBM. This transformation maintains the model performance and increases interpretability. Kim et al. [25] introduced Probabilistic Concept Bottleneck Models (ProbCBMs). ProbCBMs use probabilistic concept embeddings. They model the uncertainty in concept predictions finely. This approach offers explanations based on concepts and their uncertainties. It also significantly improves the model's robustness to input data changes.

Studies from [22–25] show that the current CBM methods focus on interpretable, image-feature-based classification. However, the current CBM methods annotate attributes with only one value. According to hematological morphology [26], blood cells have multiple attributes. These attributes often have more than one value. For example, in the same category of blood cells, the cytoplasm color may have various values. Therefore, we cannot use CBM methods alone to train on blood cell images. Instead, we need to find a method that can handle multiple values for the blood cell attribute features.

Partial label learning (PLL) is an emerging framework in weakly supervised learning [27]. It has a wide range of applications. PLL aims to learn from training samples. It associates each training sample with a set of candidate labels. Only one label is valid for the training sample, known as the ground-truth label. Using PLL can significantly reduce the time and money costs of labeling large amounts of data while ensuring model accuracy. Currently, PLL can be divided into four types: transformation strategy, theory-oriented strategy, extensions, and disambiguation strategy [28]. The transformation strategy includes binary learning, graph matching, and dictionary learning. In binary learning, Lin et al. [29] used error-correcting output codes. They combined the feature space and label space for PLL. Lyu et al. [30] proposed a PLL framework based on graph matching. They turned the one-to-one probabilistic matching algorithm into a many-to-one constraint. Chen et al. [31] proposed a dictionary-based learning method. It is for ambiguously labeled multiclass classification. They used an iterative alternating algorithm to solve the dictionary learning

problem. The transformation strategy might not solve the classification problem of blood cell attributes. Blood cells have many attributes. There are complex interactions between these attributes. The transformation strategy often simplifies the problem. It uses forms like binary learning or graph matching. This simplification cannot fully capture the subtle connections between attributes. Blood cells of the same category might have multiple correct attribute labels. The transformation strategy tends to simplify the problem's structure. This simplification might lead to the loss of key cellular attribute feature information when dealing with multi-label ambiguity.

The theory-oriented strategy focuses more on the theoretical aspects of PLL. Xu et al. [32] proposed a theoretically grounded and practically effective method. It is for instance-dependent partial label learning through Progressive Purification. Unfortunately, models in the field of medical image analysis require greater flexibility to handle challenges from data variation or uncertainty. For example, differences in the cell staining agents used by different hospitals can affect cell imaging. This can lead to model misjudgments. At the same time, theory-oriented strategies often rely on strict assumptions. This can limit the model's generalizability and adaptability.

The extensions include various semi-supervised PLL methods, like partial multi-label learning (PML) [33]. Blood cells have rich features. Each attribute has its unique candidate label range. A cell category can be defined by a combination of attributes. Thus, training on blood cell attributes to identify cell categories is essentially a multi-label partial label learning task. Unfortunately, the current PML methods are not directly suitable for learning blood cell attributes. In PML methods, the candidate labels do not point to the same attribute. For a specific attribute of blood cells, any candidate label could theoretically represent its presence.

The disambiguation strategy in PLL distinguishes the confidence associated with each candidate label to determine the ground-truth label. This strategy directly addresses the core issue of label ambiguity in partial label learning to infer the most likely correct label. Unlike the transformation strategy that simplifies the PLL problem into other machine learning problems, the disambiguation strategy seeks solutions within the original partial label framework. This approach avoids information loss or added complexity from the transformation process. The disambiguation strategy is data-centric and focuses more on experimental and practical effect verification, avoiding complex theoretical computations. Wang et al. [34] proposed the Contrastive Label Disambiguation for Partial Label Learning (PICO) algorithm. It identifies the ground-truth labels using embedded prototypes in contrastive learning. Contrast items help to obtain more precise label disambiguation prototypes. In this algorithm framework, the prototype modules play a crucial role, helping the neural network to understand and differentiate features of various cell attributes deeply. In this way, the model can match new cell samples with known blood cell attribute prototypes to accurately infer the most likely attribute values of the samples. At the same time, by using contrastive learning methods to compare the similarities and differences in the attributes between the cell samples, the neural network's ability to recognize different cell attribute features is further enhanced.

It is worth emphasizing that the blood cell dataset used in this study presents different partial label structures compared to the dataset the PICO algorithm uses. The blood cell morphology analysis [26] shows that blood cells of the same category can exhibit various attribute combinations. However, these combinations do not repeat across different blood cell categories. We have therefore preset the candidate label sets for the blood cell attribute dataset. For blood cells of the same category, we designed candidate label sets with identical structures. For blood cells of different categories, the structures of these sets vary distinctly. Each set includes the ground-truth label for a cell's attributes. In contrast, the PICO algorithm generates its candidate label sets through a random process. This approach can lead to inconsistent structures of the candidate label sets for images within the same category. In some instances, the sets may lack the ground-truth labels, as Figure 1 illustrates. In Figure 1a, within the candidate label sets for blood cell attributes, the ground-truth label

is always present. In Figure 1b, within the candidate label sets of the original method [34], the ground-truth label may not be present. Unlike the PICO algorithm, we have predefined reasonable candidate label sets for the attributes of each blood cell category, preventing the model from selecting a result blindly.

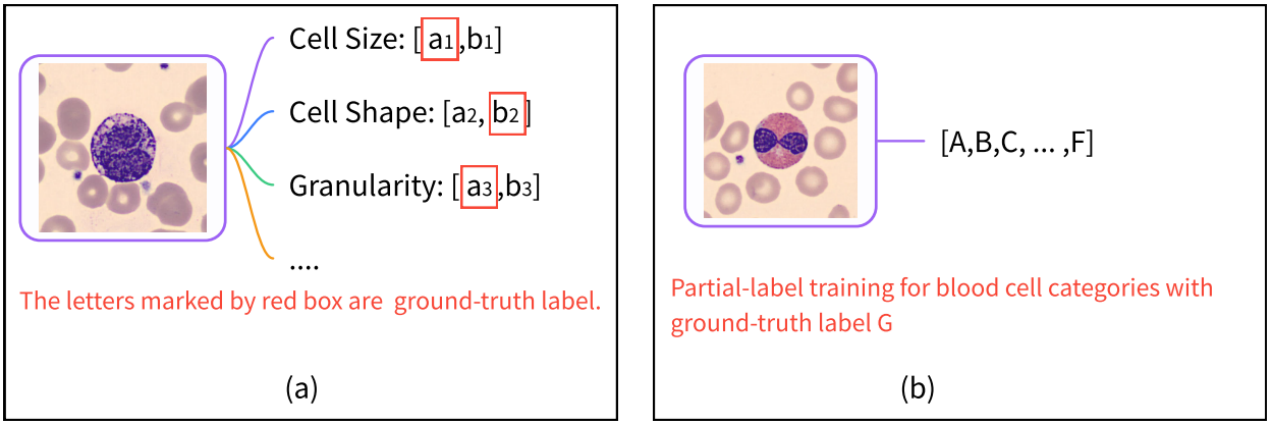

Figure 1. Comparison of partial label structures between the blood cell dataset and the PICO dataset. (**a**) Among the candidate labels for blood cell attributes, there must be a correct label; (**b**) The candidate label set in the original PICO algorithm may not contain the ground-truth label.

Therefore, this study develops a blood cell attribute classification method based on the partial label learning strategy PICO. We named this method P4BC (PLL for blood cells). This method trains deep learning networks using the partial label data of blood cell attributes. During this process, it stores the feature information of the cell attributes in prototype modules. The method uses the prediction results to calculate the attribute classification loss. It selects samples with high similarity regarding the attribute feature information stored in the prototypes as positive samples for contrastive loss calculation. This approach provides interpretability for the model's prediction results. The study employs a joint optimization strategy of these two losses. It aims to build a network model that can identify the blood cell attributes with high precision. With this attribute network model, the study further achieves the precise classification of blood cell categories. It ensures that the model can accurately determine the cell categories while maintaining high attribute classification accuracy.

This study presents three contributions.

1. We introduce the partial label learning strategy into the field of blood cell image recognition. Using this strategy, we update the attribute classification loss and contrastive loss with the feature information of the blood cell attributes stored in the prototype modules. The contrastive loss helps the model to capture the subtle differences between different cell attributes. At the same time, the attribute classification loss further guides the model to focus on and learn the key features of the attributes. The combination of these two types of losses allows the model to efficiently use the partially labeled blood cell attribute data for training. Thus, it achieves the precise prediction of blood cell attributes.
2. Based on the knowledge of blood cell morphology, we preset candidate label sets for the attributes of different blood cell categories. We integrate morphological knowledge with deep learning technology and name it "knowledge alignment". This ensures that the model makes accurate predictions based on the knowledge of blood cell morphology and the attribute feature information stored in the prototype modules during training. This not only enables the model to precisely identify the correct cell attributes within the appropriate set of attribute labels but also significantly enhances the model's interpretability. Through this method, we ensure that the model's attribute

prediction results do not conflict with the cell morphology knowledge. This reflects the scientific nature of our approach and successfully achieves knowledge alignment.

3. The method used in this study not only accurately identifies the blood cell attributes but also effectively reduces the deep learning network's dependence on a large amount of precisely annotated cell attribute data. It significantly lowers the economic and time costs of labeling. More importantly, by providing additional training for the blood cell category classification to the deep learning network, this study achieves highly precise identification of the cell categories while ensuring the accuracy of the attribute recognition. This demonstrates the great potential and significant value of our method in the practical application of blood cell attribute and category classification.

2. Materials and Methods

This section is divided into three parts. First, we introduce the dataset used in our study. Next, we describe the main body structure of the P4BC algorithm. Then, we provide the specific components and usage methods of our algorithm.

2.1. Dataset

In our study, we use the WBCAtt dataset that Tsutsui et al. [35] proposed. This dataset includes all White Blood Cells (WBCs) from the PBC dataset [36]. Pathologists, literature reviews, and manual inspection of cell images helped to annotate the cell images in the WBCAtt dataset with 11 attributes. Figure 2 displays these annotations. Red arrows in the figure serve as the classification criteria for blood cell attributes.

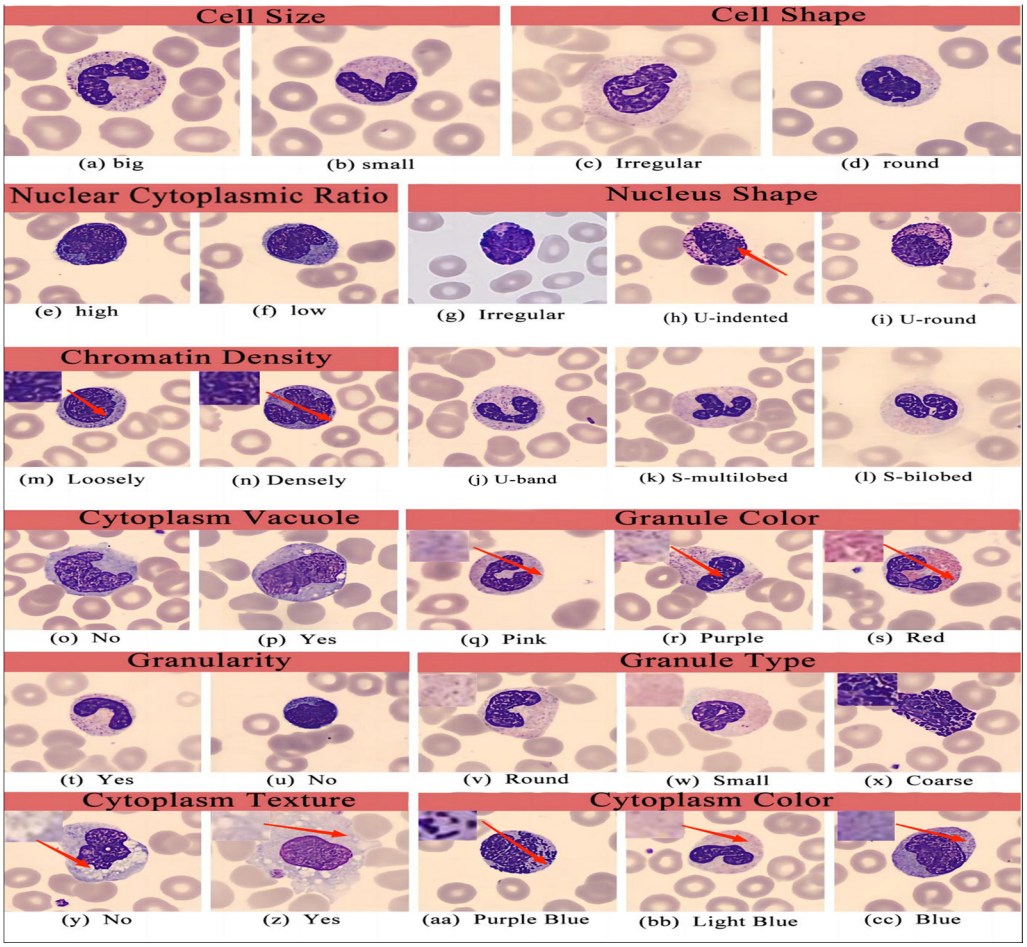

Figure 2. Eleven attributes of WBCs. "U" stands for "unsegmented," and "S" stands for "segmented".

The dataset includes 1218 basophils, 3117 eosinophils, 1420 monocytes, 3329 neutrophils, and 1214 lymphocytes. "Granule type" and "granule color" may be "nil", indicating the absence of such attributes. We have found some samples in the WBCAtt dataset that, despite belonging to different cell categories, share identical attribute annotations. Therefore, we must process the data to ensure samples of different cell categories in the entire WBC dataset have distinct attribute annotations. Our data process aims to reduce the cases of attribute combination values with very few samples.

Therefore, we propose a data processing method. This method assumes "cell image samples of different categories have different attribute value annotations". We progress through all samples one by one. We filter out pairs of samples from different categories with the same attributes (we call these "duplicate pairs") and remove them. The data processing flowchart is illustrated in Figure 3. A category of blood cells has multiple attribute combinations, with 'combination 1', 'combination 2', etc., representing different attribute combinations.

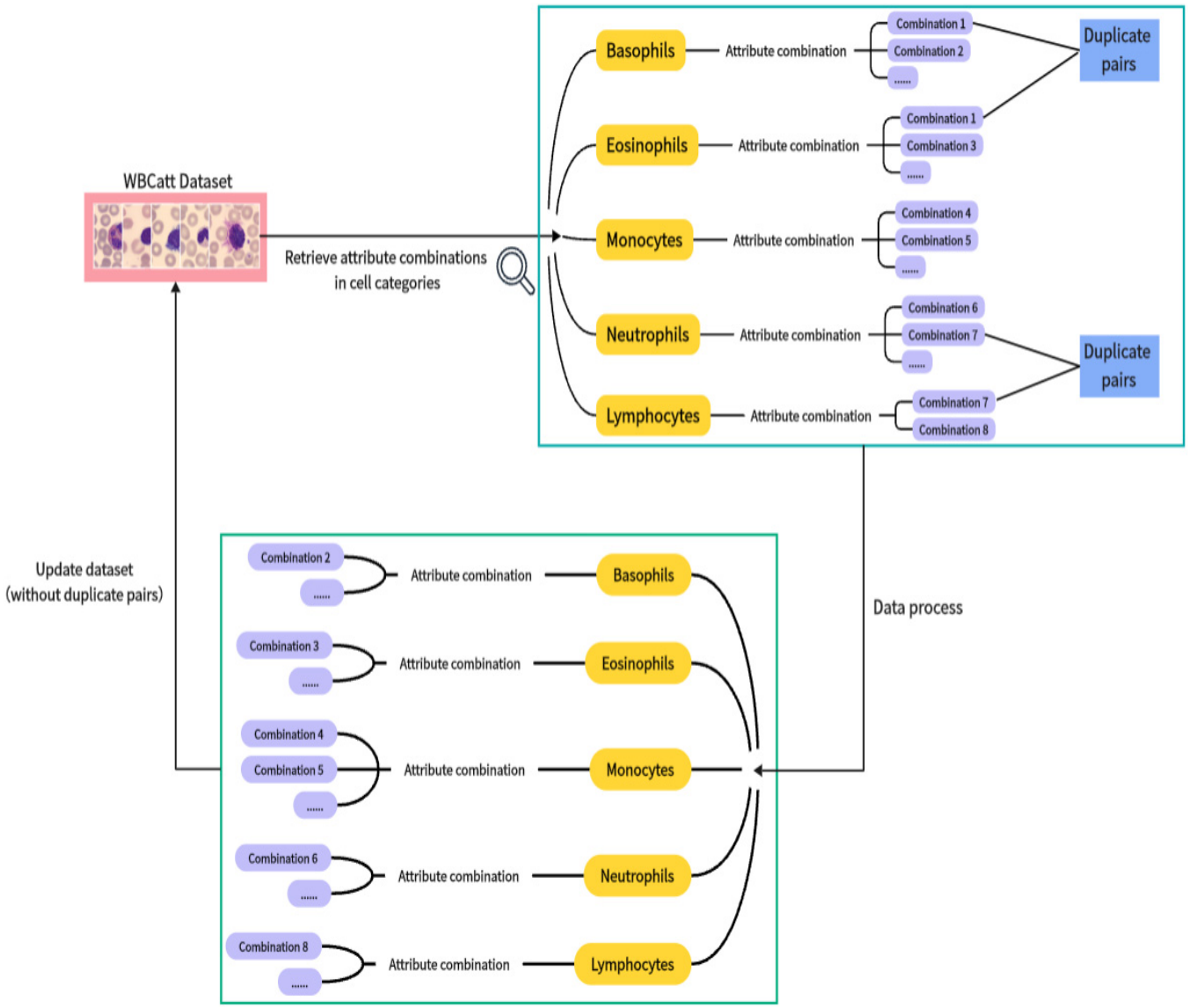

Figure 3. The data processing flowchart.

Finally, out of 10,298 samples in the WBC dataset, we filtered out 100 samples (less than 1%). This ensures no samples of different cell categories share the same attributes. After manually checking each sample, we found few samples had almost unique attribute

combination values in the entire dataset. The distribution of blood cell attributes is shown in Table 1.

Table 1. Distribution of blood cell attributes after data processing.

The Attribute Distribution of Blood Cells								
Cell size	Big	5551	**Cell shape**	Irregular	2320	**Cytoplasm vacuole**	NO	9409
	Small	4647		Round	7878		Yes	789
Nuclear cytoplasmic ratio	Low	9029	**Chromatin density**	Densely	9283	**Cytoplasm texture**	Clear	8242
	High	1169		Loosely	915		Frosted	1956
Granularity	No	2536				**Cytoplasm color**	Light blue	7794
	Yes	7662					Blue	1352
Granule color	Red	3117	**Granule type**	Small	3333		Purple blue	1052
	Nil	2537		Nil	2538	**Nucleus shape**	Unsegmented-indented	1269
	Pink	3245		Coarse	1213		Unsegmented-round	1049
	Purple	1299		Round	3114		Irregular	868
							Unsegmented band	2610
							Segmented-bilobed	4402

The distribution of blood cell categories after data cleaning is shown in Table 2.

Table 2. Distribution of blood cell categories.

Cell Category	The Number of Processed Samples
Eosinophil	3117
Basophil	1216
Lymphocyte	1178
Monocyte	1362
Neutrophil	3325
Total	10,198

Based on the knowledge of blood cell morphology [26], we considered the diverse values each blood cell attribute might have. We constructed a set of reasonable candidate label sets for each category of blood cell based on the distribution of their attribute features. This process introduces the prior knowledge of blood cell morphology to guide the subsequent neural network model training. We aim for our model to prioritize candidate values within blood cell attribute samples that align with cellular morphology, thereby diminishing the introduction of interference. This strategy is intended to reduce incidents of anomalous predictive attributes that diverge from morphological knowledge, thereby bolstering the model's interpretability and enhancing its scientific rigor. Our objective is to train a partial label learning algorithm using a weakly annotated set of blood cell attributes. We aim for this algorithm to accurately classify blood cell attributes. The predictions should conform to morphological knowledge. Table 3 shows the candidate label sets for attributes we designed under morphological guidance.

Table 3. Candidate label sets for various blood cell attributes.

		Eosinophil	**Basophil**	**Lymphocyte**	**Monocyte**	**Neutrophil**
Cell size	Big	1	1	1	1	1
	Small	1	1	1	1	1
Cell shape	Irregular	1	1	1	1	1
	Round	1	1	1	1	1

Table 3. *Cont.*

		Eosinophil	Basophil	Lymphocyte	Monocyte	Neutrophil
Nucleus shape	Unsegmented-indented	1	1	1	1	1
	Unsegmented-round	1	1	1	1	1
	Irregular	1	1	1	1	1
	Unsegmented band	1	1	0	0	1
	Segmented-bilobed	1	1	0	1	1
Nuclear cytoplasmic ratio	Low	1	1	1	1	1
	High	1	1	1	1	1
Chromatin density	Densely	1	1	1	1	1
	Loosely	1	0	1	1	1
Cytoplasm vacuole	Yes	1	1	0	1	1
	No	1	1	1	1	1
Cytoplasm texture	Clear	1	1	1	1	1
	Frosted	0	0	1	1	0
Cytoplasm color	Light blue	1	1	1	1	1
	Blue	1	0	1	1	1
	Purple blue	0	1	1	1	0
Granule type	Small	0	1	1	1	1
	Nil	1	0	1	1	1
	Coarse	1	1	1	0	1
	Round	1	0	0	0	1
Granule color	Red	1	0	0	0	1
	Nil	0	0	1	1	1
	Pink	0	0	0	1	1
	Purple	0	1	1	0	1
Granularity	No	0	0	1	1	1
	Yes	1	1	1	1	1

In the table, "1" indicates that an attribute value is included in the candidate label set and could be the ground-truth label; "0" indicates that an attribute value is not in the candidate label set and cannot be the ground-truth label.

2.2. The Overall Structure of the Algorithm

The P4BC method's structure is shown in Figure 4, consisting of two main stages.

Our method proceeds in two phases. It is crucial to note that the training image dataset remains consistent across both phases. In phase 1, we employ attribute candidate label data from the dataset. In phase 2, we use category label data from the dataset. Phase 1 focuses on the partial label learning algorithm. We employ the partial label learning algorithm to extract and store various attribute features from blood cell images, aiming to effectively address the ambiguity of partial label data. Encoders Network and Prototypes Network are crucial components of this algorithm. The Encoders Network extracts feature information from images and uses this feature information to train a classifier contained within the Encoders Network. This classifier performs predictive classification of attributes. Additionally, the feature information extracted from images by the Encoders Network is stored in the Prototypes Network. The attribute prediction results of the classifier are then used to select blood cell attributes matching the stored cell attribute feature information as positive samples for contrastive loss calculation. Additionally, these predictions are used to

compute attribute classification loss. During training, the attribute classification loss and the contrastive loss together represent the feature information of blood cell attribute data.

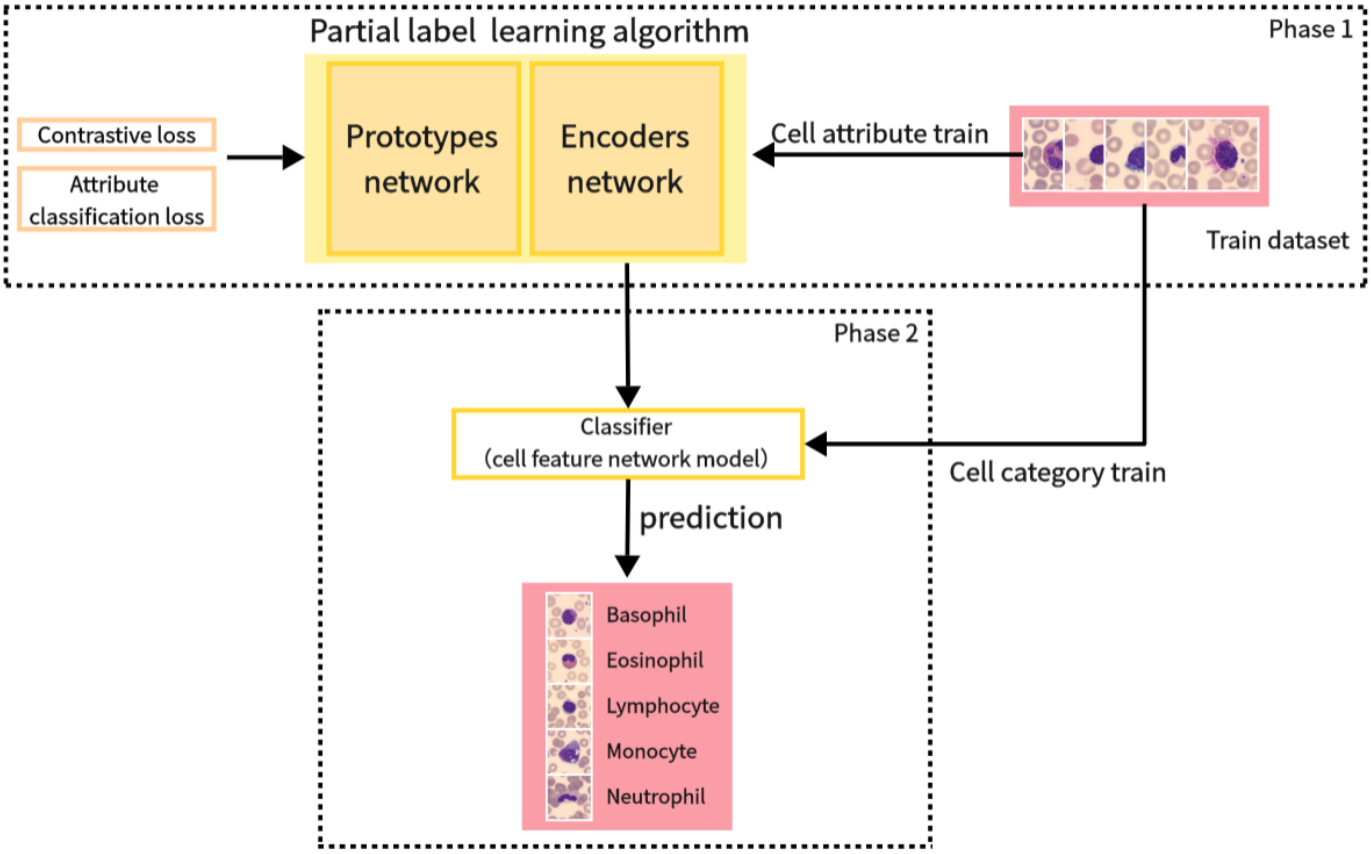

Figure 4. Structure of P4BC.

In phase 2, we use the classifier; it has comprehensively learned the feature information of blood cell attributes from phase 1's training. We refer to this as the cell features network model. Using this model, we continue training for blood cell category classification using category label data on the same training image dataset, aiming for the model to accurately predict the categories of blood cell samples.

2.3. Framework of Partial Label Learning Algorithm

Partial label learning (PLL) represents an advanced form of weak supervision, particularly well-suited for cases where blood cell attribute samples are associated with multiple candidate labels. The primary goal of PLL is to effectively train models on data with incomplete labels by employing strategies for label disambiguation, thus addressing the prevalent ambiguity issues in such datasets. In this context, our research builds upon the general principles of PLL to develop a specialized cell attribute network model. This model is designed to efficiently handle the unique challenges of blood cell image attributes, enabling precise attribute predictions. Our approach extends traditional PLL frameworks by incorporating specific enhancements tailored for high accuracy in medical imaging scenarios, as detailed in Figure 5. The term "Classifier" refers to the same concept throughout the figure. A and P are crucial elements for calculating contrastive loss. The dashed double arrows in the figure represent the mutual optimization mechanism between the loss function and prototypes.

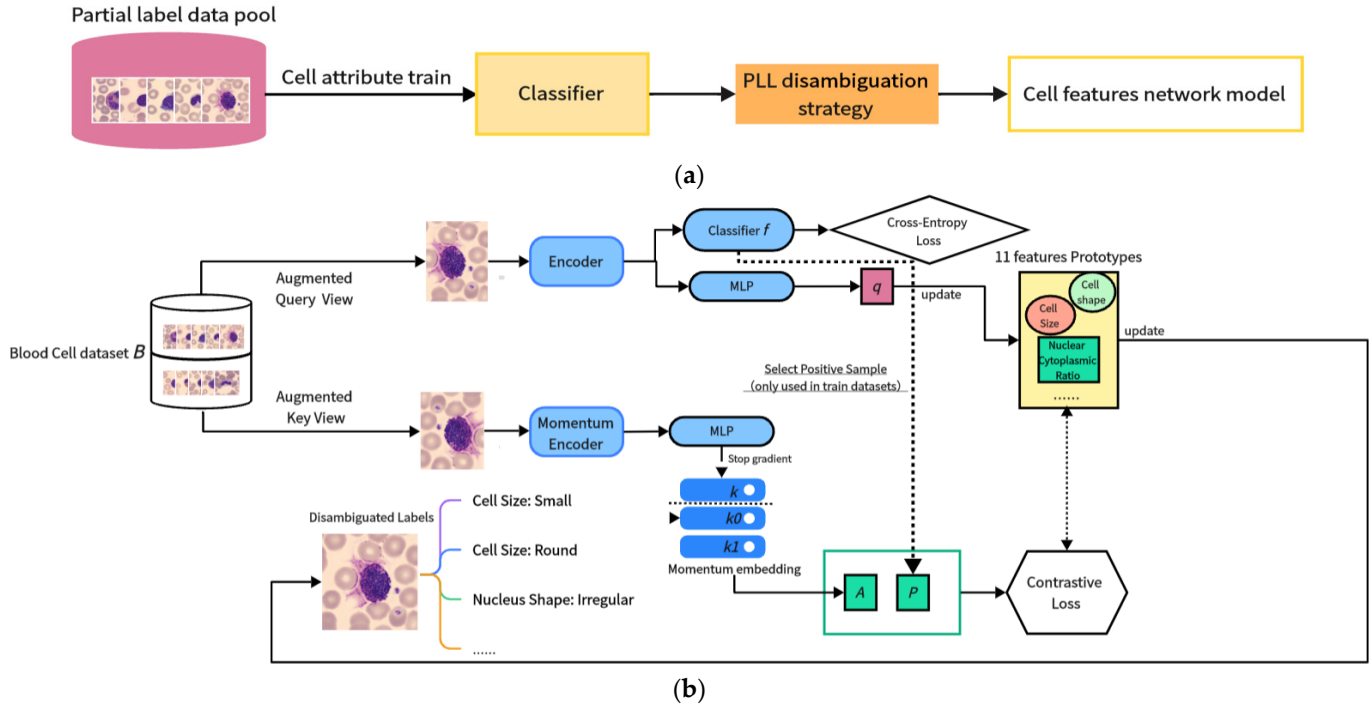

Figure 5. (**a**) Workflow of the partial label learning algorithm; (**b**) detailed explanation of the disambiguation strategy in the partial label learning algorithm.

2.3.1. Encoders Network and Prototypes Network

Before introducing the Encoders Network and Prototypes Network, we need to first explain the expression of the blood cell dataset our method uses. We define the blood cell dataset $B = \{(x_i, Y_{i,a})\}_{i=1}^{n}$. Here, n is the number of blood cell samples, x_i is a blood cell sample, and $x_{i,a}$ is an attribute of the sample. Based on blood cell morphology, we preset candidate label sets for cell attributes as $Y_{i,a} \subset \mathcal{Y} = \{1, 2, \ldots, D\}, a \in \{Cell\ shape, Cell\ size, \ldots\}$. The variable a stands for a specific cell attribute. Each attribute corresponds to a set of candidate labels.

The algorithm consists mainly of an Encoder Network and a Prototypes Network. The Encoder Network includes a Query Encoder and a Momentum Encoder. Random data augmentation techniques [37] generate two augmented samples for $(x_{i,a}, Y_a)$. These are Augmented Query View $\mathrm{Aug}_q(x_{i,a})$ and Augmented Key View $\mathrm{Aug}_k(x_{i,a})$. We then input them into the Encoders Network. $(x_{i,a}, Y_a)$ represents a blood cell attribute sample $x_{i,a}$ with a candidate label set Y_a. Aug_q, Aug_k denote data augmentation. We define a classifier $f : \mathcal{X} \to [0,1]^D$, We train the classifier using the Query View and calculate the attribute classification loss based on the prediction results.

In the MLP of the Query Encoder and Momentum Encoder, we employ the Query network $g(\cdot)$ and Key network $g'(\cdot)$ methods from the MoCo [36]. This generates a pair of L2-normalized embeddings. The Query View corresponds to Equation (1):

$$\boldsymbol{q} = g\Big(\mathrm{Aug}_q(x_{i,a})\Big) \tag{1}$$

$\boldsymbol{q}$ is used for contrastive loss calculation and participates in the update of prototypes. The Key View corresponds to Equation (2):

$$\boldsymbol{k} = g'(\mathrm{Aug}_k(x_{i,a})) \tag{2}$$

$\boldsymbol{k}$ is used for the calculation of contrastive loss.

The Prototypes Network consists of $\boldsymbol{q}$ and the prototype modules. $\boldsymbol{q}$ captures the feature information of blood cell attributes. The total number of prototype modules is

determined by the number of blood cell attributes, and each cell attribute has its set of dedicated prototype modules. Within each set of prototype modules, the number of modules directly corresponds to the possible range of attribute values. For specific attributes of a specific category of blood cells, we expect the prototypes to save only the attribute feature information that the cell category possesses according to blood cell morphology. This excludes the attribute feature information that the cell category does not possess, as shown in Equation (3):

$$\boldsymbol{\mu_c} = \text{Normalize}(\gamma \boldsymbol{u'_c} + (1-\gamma)\boldsymbol{q}), \quad \text{if } c = \underset{j \in Y_a}{\text{argmax}} f^j\left(\text{Aug}_q(x_{i,a})\right) \tag{3}$$

c represents the predicted attribute result. γ is a hyperparameter. It determines the update of prototypes using a moving average. It represents the extent of the update compared to the prototype of the previous epoch. The larger γ, the more thorough the update of the prototype, and $\boldsymbol{u'_c}$ represents the prototype modules from the previous training epoch before the current one. $c = \underset{j \in Y_a}{\text{argmax}} f^j\left(\text{Aug}_q(x_{i,a})\right)$ denotes that the classifier predicts the label index j based on the input sample $\text{Aug}_q(x_{i,a})$, within our preset candidate label set Y_a, thus not contradicting morphological knowledge. We optimize the attribute classification loss and contrastive loss using the prototype modules to disambiguate the partial labels of blood cell attributes.

2.3.2. Loss Function

Before introducing the loss function, we emphasize that, to focus the model's prediction on the most likely attribute of the current sample, we modified the concept of pseudo-labels $\boldsymbol{s}$ used in the original method [34] (in the remaining part of this article, 'original method' always represents the same meaning). $\boldsymbol{s}$ is a crucial element in calculating the loss function. For an attribute $\boldsymbol{a}$ in the blood cell sample x_i, the pseudo-label vector $\boldsymbol{s_{i,a}} \in [0,1]^D$, being updated throughout the training process, is shown in Equation (4):

$$\boldsymbol{s_{i,a,j}} = \boldsymbol{s_{i,a,j}} z_c, z_c = \begin{cases} 1 & \text{if } c = \underset{j \in Y_{i,a}}{\arg max}\ \boldsymbol{q}^\top \boldsymbol{\mu}_j \\ 0 & \text{else} \end{cases} \tag{4}$$

$\boldsymbol{\mu}_j$ represents the prototype of a blood cell attribute predicted to be index j. $\boldsymbol{s_{i,a,j}}$ denotes the pseudo-label vector for the predicted index of a blood cell attribute sample. $Y_{i,a}$ is the set of candidate labels for $x_{i,a}$. The expression $\arg \underset{j \in Y_a}{max}\ \boldsymbol{q}^\top \boldsymbol{\mu}_j$ selects the attribute c most similar to $\boldsymbol{q}$ from all possible attribute prototypes $\boldsymbol{\mu}_j$ as the predicted attribute. Here, z_c is set as a one-hot encoding where the position for attribute c is 1, and all other positions are 0.

Our PLL algorithm utilizes two loss functions: the attribute classification loss $\mathcal{L}_{cls,a}$ and the contrastive loss $\mathcal{L}_{cont,a}$. During training, the model optimizes the classifier through backpropagation of these loss functions, enhancing its understanding of blood cell attribute features. This training mechanism ensures the model gradually improves its prediction accuracy for blood cell attributes. The overall loss function is defined in Equation (5):

$$\mathcal{L}_a = \mathcal{L}_{cls,a} + \mathcal{L}_{cont,a} \tag{5}$$

$\mathcal{L}_a$ represents the loss value for a specific blood cell attribute. To optimize the classifier, we perform loss backpropagation for each attribute of the blood cell separately. Next, we will detail the design principles of attribute classification loss and contrastive loss.

The attribute classification loss updates the classifier using the cross-entropy loss function, as shown in Equation (6):

$$\mathcal{L}_{cls,a}(f; x_i, Y_{i,a}) = \sum_{j=1}^{D} -\boldsymbol{s_{i,a,j}} \log\left(f^j(x_{i,a})\right) \ s.t. \sum_{j \in Y_{i,a}} \boldsymbol{s_{i,a,j}} = 1 \tag{6}$$

$f^j(x_{i,a})$ represents the predicted index for the blood cell attribute sample $x_{i,a}$ by the classifier. The equation $\sum_{j \in Y_{i,a}} \boldsymbol{s_{i,a,j}} = 1$ indicates that only one pseudo-label vector is 1 within the candidate label set. Additionally, the algorithm employs a contrastive learning mechanism to learn different representational features under the same cell attribute. The algorithm incorporates a contrastive learning mechanism to learn different representational features under the same cell attribute. It creates a queue to store previous embeddings $\boldsymbol{k}$. Ultimately, we obtained an embedding pool for calculating contrastive loss, denoted as $A = B_q \cup B_k \cup$ queue. Here, "queue" consists of multiple sets of momentum embeddings from Figure 4, and B_q and B_k represent the current batch's embedding $\boldsymbol{q}$ and embedding $\boldsymbol{k}$, respectively. The purpose of contrastive loss is to teach the model to learn multiple cell attribute feature spaces. In these spaces, similar blood cell attribute samples become closer, while dissimilar ones are farther apart, thereby classifying the attributes of various blood cells. The classifier's prediction results are used as positive samples, as shown in Equation (7):

$$P(x_{i,a}) = \left\{ k' \middle| k' \in A(x_{i,a}), \widetilde{y_a}' = \widetilde{y_a} \right\} \tag{7}$$

$A(x_{i,a}) = A \setminus \{\boldsymbol{q}\}$, and the classifier's prediction result is $\widetilde{y_a} = \underset{j \in Y_a}{\operatorname{argmax}} f^j\left(\mathrm{Aug}_q(x_{i,a})\right)$. k' represents the embeddings in $A(x_{i,a})$ chosen for having the attribute label $\widetilde{y_a}'$ that matches the current attribute prediction result $\widetilde{y_a}$. The contrastive loss is provided in Equation (8):

$$\mathcal{L}_{cont,a}(g; x_{i,a}, \tau, A) = -\frac{1}{|P(x_{i,a})|} \sum_{\boldsymbol{k}_+ \in P(x_{i,a})} log \frac{\exp(\boldsymbol{q}^T \boldsymbol{k}_+ / \tau)}{\Sigma_{\boldsymbol{k}' \in \mathrm{A}(x_{i,a})} \exp(\boldsymbol{q}^T \boldsymbol{k}' / \tau)} v \tag{8}$$

$\tau \geq 0$ is a temperature coefficient, and $\boldsymbol{k}_+$ represents elements in the positive sample set. v is a weighting coefficient, as described in Equation (9):

$$v = \delta \boldsymbol{s_{i,a,j}} + u \; s.t. \sum_{j \in Y_{i,a}} \boldsymbol{s_{i,a,j}} = 1 \tag{9}$$

δ and u are hyperparameters. δ is used to calculate the contrastive loss when $\boldsymbol{s_{i,a,j}} = 1$. It helps the model focus on the differences in attribute feature information between the most likely correct attribute sample and the current attribute sample to optimize classifier accuracy. u is used to calculate the differences in attribute feature information between current attribute samples. It helps to reduce the impact on classifier accuracy if the wrong attribute sample is considered the most likely correct. The loss function also optimizes the prototypes. Through backpropagation of the loss, the cell attribute feature information stored in the prototypes becomes clearer and more precise.

2.3.3. The Training Process of the PLL Algorithm

Sections 2.3.1 and 2.3.2 introduce the components of our algorithm and the details of the loss functions, respectively. The following text will further describe the training process of the algorithm.

In the initial phase, we preset a reasonable set of candidate attribute labels Y_a for each cell category, based on the expertise in blood cell morphology. Data augmentation transforms the blood cell attribute sample $x_{i,a}$ into the Query View $\mathrm{Aug}_q(x_{i,a})$ and the Key View $\mathrm{Aug}_k(x_{i,a})$. We input these views into the encoder network to obtain two L2-normalized embeddings: $\boldsymbol{q}$ and $\boldsymbol{k}$, and we create a queue to store previous $\boldsymbol{k}$. We train the classifier f using the Query View and use the classifier's prediction results as positive samples. In the label prediction phase, we employ a strategy that segments attributes. Specifically, this process slices the algorithm's predicted labels, dividing the original single classification task into multiple sub-tasks. Each sub-task focuses on classifying different cell attributes. We use the index from prediction results to update the pseudo-label vector $\boldsymbol{s_{i,a,j}}$. The predicted attribute result c and $\boldsymbol{q}$ update the prototype $\boldsymbol{\mu_c}$ (initially set to zero at the

start of training). We calculate the attribute classification loss $\mathcal{L}_{cls,a}$ using $s_{i,a,j}$ and the classifier's prediction results. The contrastive loss $\mathcal{L}_{cont,a}$ is calculated using q, k, queue, and positive samples, resulting in the total loss $\mathcal{L}_a$. By sequentially backpropagating the loss for each cell attribute, it optimizes the classifier precisely, enhancing understanding of cell attribute feature information. The training process continues to refine the loss function through the ongoing update of pseudo-label vectors and the storage of attribute feature information in prototypes. Consequently, this optimized loss function, in turn, further improves the classifier's performance.

2.4. Cell Attribute Network and Cell Category Classification

Blood cell category classification is closely related to phase 2 of the P4BC framework. In phase 1 of the framework, we train the classifier located in the Encoders Network with partial label data of blood cell attributes preset based on morphological knowledge. This enables precise prediction of various blood cell attributes, resulting in a model we refer to as the cell attribute network model. Entering phase 2 of the framework, we further use this model for blood cell image category classification training. For this purpose, we conducted targeted modifications regarding the model's neural network structure, as shown in Figure 6.

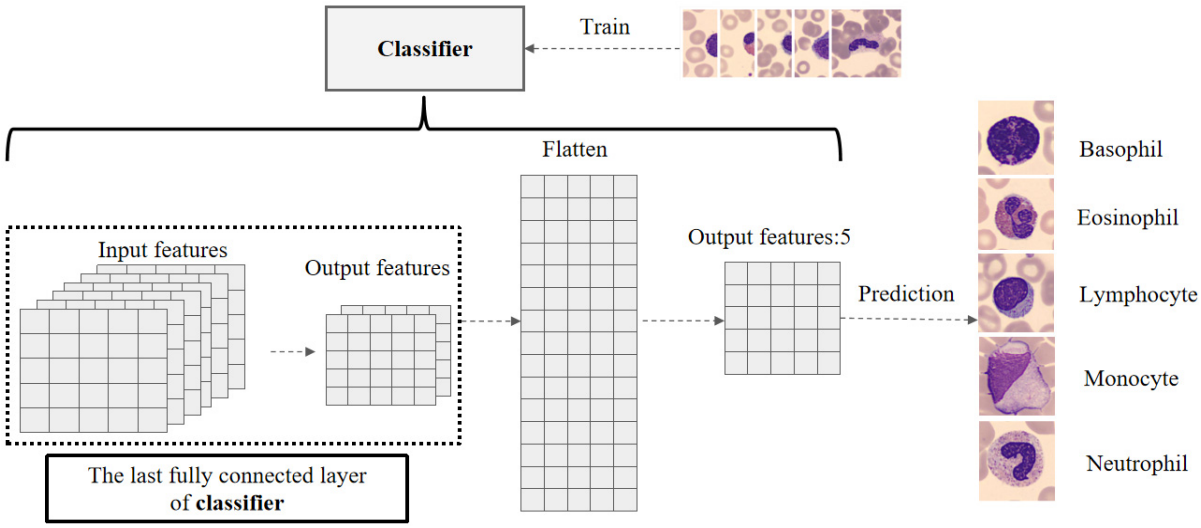

Figure 6. Modified convolutional neural network diagram.

During phase 2 of the P4BC framework training, we utilize the last fully connected layer of the classifier to unfold its output feature layer into one-dimensional data. Subsequently, these one-dimensional data are fed into an output layer with a number of output features equal to the number of cell categories, serving as the model's final prediction output. Furthermore, we continue to train the classifier f on blood cell image samples x_i optimizing the classifier using the cross-entropy loss function, as shown in Equation (10):

$$\mathcal{L}_{cls} = -\frac{1}{b}\sum_{i=1}^{b}\sum_{m=1}^{M} T_{i,m}\log(f^m(x_i)) \tag{10}$$

Here, b represents the number of samples in a batch, M represents the categories of cells, $T_{i,m}$ represents the one-hot encoded true category label vector of the blood cell image x_i, and $\log(f^m(x_i))$ denotes the probability of the sample being predicted as category m. After completing these calculations, we accumulate and average the loss for each batch to determine the overall loss. We then proceed with the backpropagation of the loss.

3. Results

This section details the experiments conducted in this study and the results obtained. We trained an attribute neural network model capable of extracting the cell attribute feature information in the first phase of P4BC (P4BC Phase1). Then, we classified the blood cell images in the second phase of the algorithm (P4BC Phase2). Additionally, we introduced evaluation metrics for attribute classification accuracy and partial label recognition accuracy. We compared the experimental results with the original method [34].

3.1. Implementation Details of the Experiment

This study utilized a NIVIDA RTX4090 graphics card with Pytorch version 1.13. In the experimental design, ResNet50 was chosen as the base network architecture, with a batch size set to 32 over a total of 500 training epochs. For the contrastive learning part, a dual-layer MLP with an output dimension of 128 served as the projection head. Additionally, to store the key vector embeddings, we established a queue with a capacity of 8192; the temperature parameter τ was set at 0.07, and the momentum coefficient for updating the Momentum Encoder was 0.99 [37]. The weight parameters δ and u were set at 0.5 and 0.1, respectively; the momentum parameter γ for prototypes updating was set at 0.99; and the update coefficient ϕ was gradually decreased from 0.95 to 0.8 [34]. These hyperparameters have been proven to be optimal. The experimental data were sourced from the pre-divided training and test sets of the WBCAtt dataset. Additionally, the data augmentation strategy followed the PICO algorithm [34].

3.2. Definition of Attribute Classification Accuracy and Partial Label Recognition Accuracy

To validate whether our model truly possesses the ability to learn from blood cell partial label data, we consider attribute classification accuracy (ACacc) and partial label recognition accuracy (PLRacc) as key metrics for measuring model performance. ACacc represents the proportion of blood cell attributes correctly identified by the model, namely the ratio of the number of correctly classified blood cell samples to the total number of samples, as shown in Equation (11):

$$\text{ACacc} = \frac{T}{N} \tag{11}$$

T is the number of correctly classified blood cell attribute samples, and N is the total number of attribute samples. PLRacc measures the model's understanding of the partial label structure and its ability to ensure prediction results always fall within the candidate label set. Taking Basophil cells as an example, suppose the candidate labels for the attribute "Granule type" are "nil", "round", "coarse". We expect the model to make a choice among these three attribute values rather than incorrectly predicting "small". The calculation of partial label recognition accuracy for a certain cell attribute is shown in Equation (12):

$$\text{PLRacc} = \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(pred_i \in Y_{i,a}) \tag{12}$$

$pred_i$ represents the model's predicted blood cell attribute result. If this prediction falls within the attribute candidate label set $Y_{i,a}$, the prediction is considered correct, yielding a result of 1; if $pred_i$ does not fall within $Y_{i,a}$, the prediction is deemed incorrect, resulting in 0.

ACacc measures the model's performance on the cell attribute classification task. A higher ACacc indicates a more thorough learning of cell attribute features by the model, demonstrating its strong capability in handling the diversity of cell attributes. PLRacc reflects the model's understanding of blood cell attribute partial label data and its ability to handle such data. The higher the PLRacc, the fewer instances where the model's predicted attribute results deviate from blood cell morphology, showing a deep comprehension of the cell attribute partial label structure by the model. Using these two ACC metrics as criteria

for measuring the model's classification performance effectively demonstrates the depth of understanding and mastery of domain knowledge after integrating blood cell morphology knowledge. The improvement in understanding not only signifies an enhancement in model interpretability but also indicates a higher quality of knowledge alignment between the morphology and deep learning technology.

3.3. Evaluation of Cell Attribute Classification Performance

In this study, we compared our proposed P4BC method with the original method in terms of blood cell attribute classification performance. The original method, as a single classification method for addressing PLL problems, trains on a specific attribute of blood cell images in each experiment. We adopted this single classification approach to independently train on 11 different blood cell attributes. Subsequently, we introduced phase 1 of the P4BC algorithm, a multiclassification strategy aimed at solving PLL problems. This algorithm allowed us to train on images of all the blood cell attributes simultaneously. Then, we evaluated the performance of these two algorithms using the ACacc and PLRacc metrics, respectively. Furthermore, to validate the necessity of each modification made to the original method, we designed a series of ablation experiments. These experiments aim to systematically assess the impact of each modification on the overall model performance, ensuring the effectiveness of our method in improving the blood cell attribute classification accuracy.

We enhanced the original method to better handle multiclassification in partial label learning. Our method integrates three key components: A, B, and C.

The original method was a single classification task for solving PLL problems, with only one group of prototype modules, and the number of modules corresponding to the number of categories. In contrast, our method is suitable for multiclassification problems in partial label learning. Considering our dataset contains 11 different blood cell attributes, we expanded the prototype modules into 11 groups, with the number of modules in each group directly reflecting the number of attribute values for the corresponding cell. Storing all the cell attribute features in one group of prototype modules might lead to confusion between attributes. We aim to reduce this risk. We saved each attribute's feature information in its respective prototype module group. We aim to improve the specificity and accuracy of the attribute classification. We named this improvement the "A" component.

The original method's candidate label set and our designed blood cell attribute candidate label set show significant structural differences. In the original method, a sample's category candidate label set might not include the sample's true label. Therefore, as the training progresses, the original method adjusts the category candidate label set, hoping to include the true label. In contrast, our attribute candidate label sets are reinforced with blood cell morphology knowledge, ensuring the sets contain the attribute's true label, and each attribute label has the potential to be the correct label. Based on this, we did not adopt the original method's pseudo-label updating mechanism but instead improved it. We hope that the attribute candidate label sets aligned with the morphology do not introduce new disturbances or lose any potential ground-truth labels during training. We named this improvement the "B" component.

In terms of the backpropagation of the loss function, the original method's strategy was to backpropagate using the average loss calculated for each batch. In contrast, our algorithm conducts separate backpropagation for the blood cell attribute loss calculated for each batch, instead of using the average value of all the blood cell attribute losses for backpropagation. We hope that the model can learn the unique feature information of each cell attribute and reduce the risk of confusing attribute features. We refer to this improvement as the "C" component.

First, we compared the original method with P4BC Phase 1 using the ACacc metric and conducted a series of ablation experiments. In these experiments, we created a version of P4BC Phase 1 from which the "A", "B", and "C" components were removed, referred

to as P without ABC. Similarly, versions removing any single component also followed a similar naming convention. The experimental results are shown in Table 4.

Table 4. Experimental results of P4BC Phase1 versus the original method for ACacc, ablation experiments of P4BC Phase1.

ACacc \ Methods	PICO	P4BC Phase1	P without C	P without BC	P without ABC
Cell size	55.32%	**55.43%**	55.47%	45.44%	52.89%
Cell shape	**77.28%**	76.25%	76.89%	42.10%	33.30%
Nucleus shape	9.80%	**11.91%**	8.46%	9.96%	15.57%
Nuclear cytoplasmic ratio	11.46%	**87.20%**	12.87%	16.84%	69.47%
Chromatin density	90.82%	90.77%	**90.85%**	37.05%	13.77%
Cytoplasm vacuole	92.02%	**92.02%**	7.98%	85.11%	42.85%
Cytoplasm texture	80.08%	**80.12%**	80.08%	78.71%	25.78%
Cytoplasm color	75.55%	75.52%	**75.55%**	10.71%	12.04%
Granule type	25.39%	25.39%	25.75%	**26.35%**	25.00%
Granule color	12.18%	**30.17%**	25.52%	25.75%	31.72%
Granularity	74.38%	**74.64%**	74.64%	74.64%	74.64%
Average ACacc	54.93%	**63.59%**	48.55%	41.09%	36.24%

Compared to the original method, our method shows an 8.66% increase in average ACacc. The performance for "Nuclear cytoplasmic ratio" notably increases by 75.74%. "Granule color" also rises by 17.99%. These results confirm the accuracy of the prototype modules in our algorithm in distinguishing and storing feature information for cell attributes. Updating pseudo-labels without changing our attribute candidate label set structure allows the model to focus on the most probable attribute results. These results align with the blood cell morphology. Independently backpropagating the loss for each blood cell attribute reduces the risk of mixing attribute feature information. This process further boosts the model's learning efficiency. These results demonstrate that our algorithm surpasses the original method in classifying specific cell attributes, proving its superiority in blood cell attribute classification. The method proposed in this study is a partial label multiclassification algorithm. It trains all the blood cell attribute samples within a single deep learning neural network. The original method functions as a single-classification model, dedicating each iteration to training one cell attribute and requiring a distinct model for training each attribute. In contrast, our algorithm can train multiple attributes simultaneously within a single model. This integrated method surpasses the original in cell attribute classification and demonstrates our algorithm's efficiency.

The ablation experiments show that each modification we made significantly affects the blood cell attribute classification performance of P4BC Phase1. Removing the "C" component led to a 15.04% decrease in the average ACacc. Notably, the overall attribute classification performance suffered greatly under the condition of removing the "C" component (i.e., P without C). This result indicates that, while backpropagating the average value of blood cell attribute losses might slightly enhance the model's ability to recognize and classify specific attributes locally, it significantly reduces the model's accuracy in classifying other cell attributes. This phenomenon suggests that backpropagating the average value of blood cell attribute losses causes a dispersion of model attention. This dispersion inevitably integrates feature information from other attributes when learning the features of a specific cell attribute, thus increasing the risk of confusion in feature recognition and significantly weakening the overall learning efficiency of the model. In the scenario of P without C after removing the "B" component, we observed a 7.46% decrease in ACacc. This result indicates that continuing with the original method's pseudo-label strategy led to changes in our

predefined candidate label set structures, reducing the model's classification performance in later training. This decrease in performance could be due to the incorrect replacement of true labels for blood cell attribute samples in the sets or the introduction of options into the candidate label set for the specific categories of blood cell attributes that do not align with the morphology. This further hindered the model's ability to effectively learn the attributes of certain categories of cells based on morphology. This discovery emphasizes the scientific nature and importance of attribute candidate label sets based on morphological knowledge. It also highlights the necessity of aligning domain knowledge with deep learning technology. Removing the "A" component, features of different blood cell attributes were stored in the same prototype modules. This approach hinders the model's ability to distinguish between attribute features, leading to a disorganized state of feature information in the prototypes. As a result, this confusion has serious consequences, affecting the model's performance and accuracy.

3.4. Partial Label Recognition Performance Evaluation

We used PLRacc as an evaluation metric to compare the performance of our algorithm with the original method, supplemented by ablation experiments for analysis. The results are shown in Table 5.

Table 5. Experimental results of P4BC Phase1 versus the original algorithm for PLRacc, ablation experiments of P4BC Phase1.

Average PLRacc for Cell Attributes					
Methods	**PICO**	**P4BC Phase1**	**P without C**	**P without BC**	**P without ABC**
Average Accuracy	91.56%	92.65%	**94.08%**	88.21%	77.57%

P4BC Phase1 achieved a 1.09% improvement over the original method. Thus, our method shows superior performance and clear advantages in handling blood cell attribute partial label data. In our study, comparing P4BC Phase1 to the P without C strategy, which averages all the cell attribute losses before backpropagation, showed a 1.43% increase. This indicates that averaging the attribute losses for backpropagation might blur the distinctions between different attributes. However, it effectively shifts the model's focus from complex individual attributes to overall attributes. This shift allows for a more balanced understanding of each attribute and its candidate label set. However, a significant downside of P without C is its reduced focus on attributes, which should be prioritized. This reduction leads to a decline in the overall attribute classification performance. Therefore, in our study, we adopted the P4BC Phase1 strategy to balance PLRacc and ACacc.

Our algorithm surpasses the original method in learning the blood cell attribute features and understanding the partial label structures. This indicates the effective application of partial label structures built on blood cell morphology knowledge in the training process of deep learning networks. It ensures the accurate preservation of the cell attributes in the prototype modules that align with the morphology. Also, the pseudo-label updates and prototype modules interact during the optimization of the loss function. This interaction continuously guides the model to precisely learn the blood cell attribute features and distinguish between different attribute features. This method enables the model to deeply understand the feature information of each blood cell attribute and accurately identify the true label among multiple potential attribute labels. This study not only validates the effectiveness of integrating blood cell morphology knowledge into the deep learning training process but also achieves the alignment of domain knowledge with deep learning technology. By using prototype modules that store blood cell attribute feature information and generating predictions that align with blood cell morphology, the model's interpretability is significantly enhanced.

3.5. Cell Category Classification Based on Blood Cell Attribute Network Model

In phase 1 of the P4BC framework, we trained a blood cell attribute network model using blood cell attribute partial label data. In phase 2 of the framework, this network

is used to perform the blood cell category classification task. We conducted a detailed comparison of its performance with a ResNet50 model that was not specifically trained on blood cell attributes. Recall is provided by Equation (10):

$$\text{Recall} = \frac{TP}{TP+FN} \tag{13}$$

where TP represents the number of correct positive predictions by the model. FN represents the number of instances where the model incorrectly predicts a positive as negative. Precision is provided by Equation (11):

$$\text{Precision} = \frac{TP}{TP+FP} \tag{14}$$

where FP represents the number of false positives where the model incorrectly predicts a negative as positive. Comparative results are shown in Table 6.

Table 6. Category classification results of blood cell images.

	Model	Accuracy	Precision	Recall
Basophil	Cell attribute network	99.71%	96.13%	99.71%
	ResNet50	99.71%	**97.21%**	99.71%
Eosinophil	Cell attribute network	**99.57%**	99.78%	**99.57%**
	ResNet50	99.14%	**100.00%**	99.14%
Lymphocyte	Cell attribute network	97.55%	**99.17%**	97.55%
	ResNet50	**97.82%**	98.63%	**97.82%**
Monocyte	Cell attribute network	97.53%	97.75%	97.53%
	ResNet50	**97.98%**	**97.98%**	**97.98%**
Neutrophil	Cell attribute network	98.71%	99.31%	98.71%
	ResNet50	**99.51%**	99.31%	**99.51%**
Average	Cell attribute network	98.77%	98.39%	98.61%
	ResNet50	**99.00%**	**98.62%**	**98.83%**

The data show that the cell attribute network model and ResNet50 perform similarly in cell classification accuracy, precision, and recall, with the largest gap not exceeding 0.23%. This gap is within an acceptable error margin. It ensures the recognition accuracy of blood cell attributes while maintaining high accuracy for blood cell categories.

4. Discussion

In the field of medical image processing, accurately classifying the specific categories of cells is crucial for doctors. Specific blood cells show attributes that are hard to separate due to their morphology. Thus, creating a deep learning network for blood cell classification requires extensive accurately labeled data, incurring high costs. We propose P4BC, a partial label learning algorithm to train networks with less precisely labeled data, aiming for accurate blood cell attribute classification while cutting costs. The model focuses on the likely attribute labels using a pseudo-label update mechanism and stores the attribute features in prototype modules. By optimizing the classification and contrastive losses and backpropagating each attribute's loss independently, our approach prevents feature confusion, enhancing the understanding of attribute nuances.

Our algorithm outperforms the original method with an 8.66% increase in blood cell attribute recognition accuracy and a 1.09% rise in partial label recognition accuracy. These enhancements underscore our method's effectiveness in classifying blood cell attributes and understanding partial label structures. Our candidate label sets differ from the original method's randomly generated candidate label sets by including morphological knowledge. This ensures that the reasonable attribute ground-truth labels are always in the sets. This integration boosts the interpretability and accuracy of our model, offering more reliable predictions based on morphological insights. Our algorithm trains multiple cell

attributes together in a single model, marking a substantial efficiency gain over the original method. The original method demands a unique model for each attribute's training. Our unified training method not only enhances the cell attribute classification accuracy but also demonstrates our algorithm's heightened efficiency. Our approach not only maintains high accuracy in attribute prediction but also achieves the knowledge alignment between deep learning and blood cell morphology, while ensuring high precision in cell category classification as well.

Our algorithm has its limitations. First, the PLRacc metric shows a slight decrease for P4BC Phase1 compared to P without C. This forces our algorithm to find a balance between PLRacc and ACacc, ensuring both an understanding of the cell attribute partial label structure and the ability to learn attribute features. Therefore, we chose P4BC as our final approach. Our cell attribute network model shows a slight decrease in accuracy, precision, and recall rates compared to ResNet50 trained without attribute partial labels. This may be due to the increased complexity introduced by the attribute partial labels. Moreover, these outcomes suggest the need for the further optimization of the model structure or training strategies in future research. This would allow the algorithm to better utilize partial label data, enhancing both the understanding of the partial label structure and the learning of attribute features, and thereby improving the cell category classification performance. Despite these challenges, partial label learning can reduce the cost and time associated with labeling cell attribute data while achieving an excellent attribute classification performance. Partial label learning based on blood cell attributes remains a promising research direction.

5. Conclusions

In this study, we introduced the P4BC framework, a partial label learning method designed to address the high cost of attribute annotation for blood cell images in the medical field. We built candidate label sets for blood cell attributes based on professional knowledge of the blood cell morphology. The goal was to train a deep learning network to extensively understand the feature information of blood cells. This network can accurately predict the specific attributes of blood cells from a scientifically reasonable set of candidate labels. This approach has effectively integrated blood cell morphology knowledge into the model training process. It achieves an alignment between disciplinary knowledge and deep learning technology. It also ensures the interpretability and scientific validity of the prediction results. Moreover, our method has shown high accuracy in cell category recognition. This demonstrates our method's strong potential and significant advantages in classifying cell attributes and cell categories.

Author Contributions: Conceptualization, J.F. and Q.G.; methodology, J.F. and Q.M.; formal analysis, S.L.; investigation, L.C.; data curation, L.C.; writing—original draft preparation, J.F.; writing—review and editing, J.F. and Q.M.; visualization, Q.G.; supervision, Q.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Guangdong Basic and Applied Basic Research Foundation (2023A1515012966), the Special Funds for the Cultivation of Guangdong College Students' Scientific and Technological Innovation ("Climbing Program" Special Funds) (No. pdjh2023b0142), the Key-Area Research and Development Program of Guangdong Province (2020B090922006), and the Key field research projects in Foshan City (Grant No. 2120001009232).

Data Availability Statement: Due to privacy restrictions, research data in manuscripts cannot be disclosed.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Merino, A.; Puigví, L.; Boldú, L.; Alférez, S.; Rodellar, J. Optimizing Morphology through Blood Cell Image Analysis. *Int. J. Lab. Hematol.* **2018**, *40*, 54–61. [CrossRef]
2. Yao, K.; Rochman, N.D.; Sun, S.X. Cell Type Classification and Unsupervised Morphological Phenotyping From Low-Resolution Images Using Deep Learning. *Sci. Rep.* **2019**, *9*, 13467. [CrossRef] [PubMed]

3. Khan, S.; Sajjad, M.; Hussain, T.; Ullah, A.; Imran, A.S. A Review on Traditional Machine Learning and Deep Learning Models for WBCs Classification in Blood Smear Images. *IEEE Access* **2021**, *9*, 10657–10673. [CrossRef]
4. Kilicarslan, S.; Celik, M.; Sahin, Ş. Hybrid Models Based on Genetic Algorithm and Deep Learning Algorithms for Nutritional Anemia Disease Classification. *Biomed. Signal Process. Control* **2021**, *63*, 102231. [CrossRef]
5. Miikkulainen, R.; Liang, J.; Meyerson, E.; Rawal, A.; Fink, D.; Francon, O.; Raju, B.; Shahrzad, H.; Navruzyan, A.; Duffy, N.; et al. 14—Evolving Deep Neural Networks. In *Artificial Intelligence in the Age of Neural Networks and Brain Computing*, 2nd ed.; Kozma, R., Alippi, C., Choe, Y., Morabito, F.C., Eds.; Academic Press: Cambridge, MA, USA, 2024; pp. 269–287. ISBN 978-0-323-96104-2.
6. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
7. Zhou, T.-Y.; Huo, X. Learning Ability of Interpolating Deep Convolutional Neural Networks. *Appl. Comput. Harmon. Anal.* **2024**, *68*, 101582. [CrossRef]
8. Xu, J.; Liu, Y.; Fang, M. A RAW Image Noise Suppression Method Based on BlockwiseUNet. *Electronics* **2023**, *12*, 4346. [CrossRef]
9. Kim, J.; Kim, J. Augmenting Beam Alignment for mmWave Communication Systems via Channel Attention. *Electronics* **2023**, *12*, 4318. [CrossRef]
10. Küttel, D.; Kovács, L.; Szölgyén, Á.; Paulik, R.; Jónás, V.; Kozlovszky, M.; Molnár, B. Artifact Augmentation for Enhanced Tissue Detection in Microscope Scanner Systems. *Sensors* **2023**, *23*, 9243. [CrossRef]
11. Li, J.; Liang, W.; Yin, X.; Li, J.; Guan, W. Multimodal Gait Abnormality Recognition Using a Convolutional Neural Network–Bidirectional Long Short-Term Memory (CNN-BiLSTM) Network Based on Multi-Sensor Data Fusion. *Sensors* **2023**, *23*, 9101. [CrossRef]
12. Acevedo, A.; Alférez, S.; Merino, A.; Puigví, L.; Rodellar, J. Recognition of Peripheral Blood Cell Images Using Convolutional Neural Networks. *Comput. Methods Programs Biomed.* **2019**, *180*, 105020. [CrossRef] [PubMed]
13. Kutlu, H.; Avci, E.; Özyurt, F. White Blood Cells Detection and Classification Based on Regional Convolutional Neural Networks. *Med. Hypotheses* **2020**, *135*, 109472. [CrossRef] [PubMed]
14. Qadri, A.M.; Raza, A.; Eid, F.; Abualigah, L. A Novel Transfer Learning-Based Model for Diagnosing Malaria from Parasitized and Uninfected Red Blood Cell Images. *Decis. Anal. J.* **2023**, *9*, 100352. [CrossRef]
15. Lu, N.; Min Tay, H.; Petchakup, C.; He, L.; Gong, L.; Khine Maw, K.; Yuan Leong, S.; Wei Lok, W.; Boon Ong, H.; Guo, R.; et al. Label-Free Microfluidic Cell Sorting and Detection for Rapid Blood Analysis. *Lab Chip* **2023**, *23*, 1226–1257. [CrossRef] [PubMed]
16. Liao, Z.; Zhang, Y.; Li, Z.; He, B.; Lang, X.; Liang, H.; Chen, J. Classification of Red Blood Cell Aggregation Using Empirical Wavelet Transform Analysis of Ultrasonic Radiofrequency Echo Signals. *Ultrasonics* **2021**, *114*, 106419. [CrossRef]
17. Balasubramanian, K.; Ananthamoorthy, N.P.; Ramya, K. An Approach to Classify White Blood Cells Using Convolutional Neural Network Optimized by Particle Swarm Optimization Algorithm. *Neural Comput Applic* **2022**, *34*, 16089–16101. [CrossRef]
18. Khan, S.; Sajjad, M.; Abbas, N.; Escorcia-Gutierrez, J.; Gamarra, M.; Muhammad, K. Efficient Leukocytes Detection and Classification in Microscopic Blood Images Using Convolutional Neural Network Coupled with a Dual Attention Network. *Comput. Biol. Med.* **2024**, *174*, 108146. [CrossRef] [PubMed]
19. Jung, J.; Garnett, E.; Vispo, B.; Chen, X.; Cao, J.; Tarek Elghetany, M.; Devaraj, S. Misidentification of Unstable, Low Oxygen Affinity Hemoglobin Variant. *Clin. Chim. Acta* **2020**, *509*, 177–179. [CrossRef] [PubMed]
20. Razzak, M.I.; Naz, S.; Zaib, A. Deep Learning for Medical Image Processing: Overview, Challenges and the Future. In *Classification in BioApps: Automation of Decision Making*; Dey, N., Ashour, A.S., Borra, S., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 323–350. ISBN 978-3-319-65981-7.
21. Koh, P.W.; Nguyen, T.; Tang, Y.S.; Mussmann, S.; Pierson, E.; Kim, B.; Liang, P. Concept Bottleneck Models. In Proceedings of the 37th International Conference on Machine Learning, Virtual Event, 13–18 July 2020; pp. 5338–5348.
22. Zarlenga, M.E.; Barbiero, P.; Ciravegna, G.; Marra, G.; Giannini, F.; Diligenti, M.; Precioso, F.; Melacci, S.; Weller, A.; Lio, P.; et al. Concept Embedding Models. In Proceedings of the NeurIPS 2022—36th Conference on Neural Information Processing Systems, New Orleans, LO, USA, 28 November–9 December 2022.
23. Yuksekgonul, M.; Wang, M.; Zou, J. Post-Hoc Concept Bottleneck Models. *arXiv* **2022**, arXiv:2205.15480.
24. Kim, E.; Jung, D.; Park, S.; Kim, S.; Yoon, S. Probabilistic Concept Bottleneck Models. *arXiv* **2023**, arXiv:2306.01574.
25. Hoffbrand, V.; Vyas, P.; Campo, E.; Haferlach, T.; Gomez, K. Molecular Biology of the Cell. In *Color Atlas of Clinical Hematology*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2018; pp. 1–26. ISBN 978-1-119-17065-5.
26. Wu, W.; Liao, S.; Lu, Z. White Blood Cells Image Classification Based on Radiomics and Deep Learning. *IEEE Access* **2022**, *10*, 124036–124052. [CrossRef]
27. Xu, N.; Lv, J.; Geng, X. Partial Label Learning via Label Enhancement. In Proceedings of the AAAI Conference on Artificial Intelligence, 17 July 2019; Volume 33, pp. 5557–5564.
28. Tian, Y.; Yu, X.; Fu, S. Partial Label Learning: Taxonomy, Analysis and Outlook. *Neural Netw.* **2023**, *161*, 708–734. [CrossRef] [PubMed]
29. Lin, G.-Y.; Xiao, Z.-Y.; Liu, J.-T.; Wang, B.-Z.; Liu, K.-H.; Wu, Q.-Q. Feature Space and Label Space Selection Based on Error-Correcting Output Codes for Partial Label Learning. *Inf. Sci.* **2022**, *589*, 341–359. [CrossRef]
30. Lyu, G.; Feng, S.; Wang, T.; Lang, C.; Li, Y. GM-PLL: Graph Matching Based Partial Label Learning. *IEEE Trans. Knowl. Data Eng.* **2021**, *33*, 521–535. [CrossRef]
31. Chen, Y.-C.; Patel, V.M.; Chellappa, R.; Phillips, P.J. Ambiguously Labeled Learning Using Dictionaries. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 2076–2088. [CrossRef]

32. Xu, N.; Liu, B.; Lv, J.; Qiao, C.; Geng, X. Progressive Purification for Instance-Dependent Partial Label Learning. In Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; pp. 38551–38565.
33. Xie, M.-K.; Huang, S.-J. Partial Multi-Label Learning. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32. [CrossRef]
34. Wang, H.; Xiao, R.; Li, Y.; Feng, L.; Niu, G.; Chen, G.; Zhao, J. PiCO: Contrastive Label Disambiguation for Partial Label Learning. In Proceedings of the Tenth International Conference on Learning Representations, ICLR 2022, Virtual, 25–29 April 2022.
35. Tsutsui, S.; Pang, W.; Wen, B. WBCAtt: A White Blood Cell Dataset Annotated with Detailed Morphological Attributes. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 50796–50824.
36. Acevedo, A.; Merino, A.; Alférez, S.; Molina, Á.; Boldú, L.; Rodellar, J. A Dataset of Microscopic Peripheral Blood Cell Images for Development of Automatic Recognition Systems. *Data Brief* **2020**, *30*, 105474. [CrossRef]
37. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum Contrast for Unsupervised Visual Representation Learning. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 19 June 2020; pp. 9726–9735.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.