machine learning & knowledge extraction

MDPI

Article

Towards Self-Conscious AI Using Deep ImageNet Models: Application for Blood Cell Classification

Mohamad Abou Ali [1,2], Fadi Dornaika [1,3,*] and Ignacio Arganda-Carreras [1,3,4,5]

[1] Department of Computer Science and Artificial Intelligence, University of the Basque Country (UPV/EHU), Manuel Lardizabal, 1, 20018 San Sebastian, Spain; mohamad.abouali01@liu.edu.lb (M.A.A.); ignacio.arganda@ehu.eus (I.A.-C.)
[2] Department of Biomedical Engineering, Lebanese International University (LIU), Salim Salam, Mouseitbeh, Beirut 146404, Lebanon
[3] IKERBASQUE, Basque Foundation for Science, Plaza Euskadi, 5, 48009 Bilbao, Spain
[4] Donostia International Physics Center (DIPC), Manuel Lardizabal, 4, 20018 San Sebastian, Spain
[5] Biofisika Institute (CSIC, UPV/EHU), Barrio Sarriena s/n, 48940 Leioa, Spain
* Correspondence: fadi.dornaika@ehu.eus

Abstract: The exceptional performance of ImageNet competition winners in image classification has led AI researchers to repurpose these models for a whole range of tasks using transfer learning (TL). TL has been hailed for boosting performance, shortening learning time and reducing computational effort. Despite these benefits, issues such as data sparsity and the misrepresentation of classes can diminish these gains, occasionally leading to misleading TL accuracy scores. This research explores the innovative concept of endowing ImageNet models with a self-awareness that enables them to recognize their own accumulated knowledge and experience. Such self-awareness is expected to improve their adaptability in various domains. We conduct a case study using two different datasets, PBC and BCCD, which focus on blood cell classification. The PBC dataset provides high-resolution images with abundant data, while the BCCD dataset is hindered by limited data and inferior image quality. To compensate for these discrepancies, we use data augmentation for BCCD and undersampling for both datasets to achieve balance. Subsequent pre-processing generates datasets of different size and quality, all geared towards blood cell classification. We extend conventional evaluation tools with novel metrics—"accuracy difference" and "loss difference"—to detect overfitting or underfitting and evaluate their utility as potential indicators for learning behavior and promoting the self-confidence of ImageNet models. Our results show that these metrics effectively track learning progress and improve the reliability and overall performance of ImageNet models in new applications. This study highlights the transformative potential of turning ImageNet models into self-aware entities that significantly improve their robustness and efficiency in various AI tasks. This groundbreaking approach opens new perspectives for increasing the effectiveness of transfer learning in real-world AI implementations.

Keywords: self-aware AI; blood cell classification; convolutional neural networks (CNNs); ImageNet models; deep learning (DL); transfer learning (TL)

check for updates

Citation: Abou Ali, M.; Dornaika, F.; Arganda-Carreras, I. Towards Self-Conscious AI Using Deep ImageNet Models: Application for Blood Cell Classification. *Mach. Learn. Knowl. Extr.* **2024**, *6*, 2400–2421. https://doi.org/10.3390/make6040118

Academic Editor: Andreas Holzinger

Received: 22 August 2024
Revised: 20 September 2024
Accepted: 23 September 2024
Published: 21 October 2024

1. Introduction

The exploration of self-consciousness and self-awareness in artificial intelligence (AI) [1–3] involves a profound understanding of individuality, emotions, motivations and the subtleties of behavior. This heightened cognitive state enables entities to make informed decisions, set meaningful goals and adapt dynamically. A crucial question arises: can AI models achieve a comparable level of cognitive sophistication for making informed decisions?

To answer this question, it is important to explore the basic principles and evolution of AI. The rise of AI, which is often compared with modern electricity due to its transformative potential, is due to machine learning (ML), the fundamental sub-discipline of AI. The emergence of deep learning (DL) within ML has catapulted AI to unprecedented heights, particularly in image classification, where the task is to categorize images within a predefined set of possibilities.

Figure 1 shows representative images of peripheral blood smears with eosinophils, a type of white blood cell (WBC) [4,5]. In particular, Figure 1a shows an eosinophil from the PBC dataset, which is known for its high-quality and rich WBC images [4]. In contrast, Figure 1b shows an eosinophil image from the BCCD dataset, which has lower image quality and fewer WBC images compared with the PBC dataset [5]. Figure 1c highlights the background of the PBC image (Figure 1a) on the left and the eosinophilic cell on the right [4]. Figure 1d further separates the eosinophilic cell into its nucleus and cytoplasm [4]. The structure and size of the nucleus and cytoplasm help hematologists and AI models to distinguish eosinophils from other WBCs.

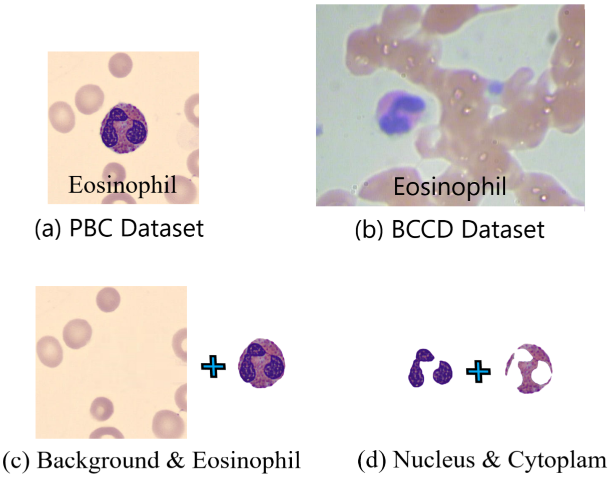


Figure 1. Representative images of peripheral blood smears with eosinophils: (**a**) eosinophil from the PBC dataset, characterized by high-quality imaging; (**b**) eosinophil from the BCCD dataset, which has lower image quality; (**c**) the PBC image's background on the left and eosinophil cell on the right; (**d**) the separation of the eosinophil's nucleus and cytoplasm.

Traditional methods for classifying digital neutrophils involve developing a feature extraction algorithm and passing these features to a classifier for eosinophil recognition. This process requires the creation of a feature extractor and classifier using advanced image processing techniques. In machine learning, complicated image processing techniques are used to create a feature extractor that serves as the input to ML classifiers such as XGBoost, random forests (RFs) and support vector machines (SVMs). In contrast, deep learning seamlessly automates both feature extraction and classification [6,7]. Deep learning, especially through convolutional neural networks (CNNs), uses convolutional filters as feature extractors and fully connected (FC) dense layers as classifiers [8,9]. These large-scale CNNs, often referred to as ConvNets, require a large dataset, such as the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) dataset, which contains 1,281,167 training images, 50,000 validation images and 100,000 test images [10]. The ILSVRC's goal, achieved in 2015 with the ResNet-50 model with an error rate of 3.6%, underscores the goal of cultivating ConvNets with an error rate below 5%, which corresponds to the average human error [11]. This success has highlighted the suitability of ImageNet models for various image feature extraction and classification tasks and has stimulated research into transfer learning (TL) [12].

There is a critical challenge in transfer learning—the lack of a standardized understanding of the required dataset size for effective TL processes. This knowledge gap impairs the achievement of exceptional accuracy, with researchers attributing inaccuracies to inadequate TL dataset sizes [13]. This paper attempts to address this challenge by investigating the feasibility of building self-aware ImageNet models.

After this introduction in Section 1, this paper is structured as follows: Section 2 reviews the relevant literature on the feasibility of self-learning AI models. Section 3 outlines the materials and methods used in the study. Section 4 presents the experimental results. Section 5 provides a discussion of these results, and Section 6 concludes the study.

2. Related Works

This work deals with the further development of self-aware ImageNet models, a relatively unexplored area in AI research. There is little direct literature on the translation of ImageNet models into conscious entities. However, there are a few studies looking at the future possibilities of conscious machines. One study investigates whether computers can become conscious and surpass human intelligence [14]. Another work explores the potential of developing conscious machines in the future [15]. Another approach is presented in research on data-aware deep learning models for product classification [16]. In addition, an article discusses whether machine understanding requires consciousness [17].

2.1. Transfer Learning (TL)

Transfer learning (TL) is a widely used practice in computer vision in which a model trained for one task is adapted for a different but related task. This method uses existing models so that there is no need to start from scratch, and high performance is achieved with smaller datasets and less computing power [18,19].

With TL, a base network is first trained on a source dataset and a task. The learned features are then transferred to a target model, which is retrained with a target dataset and a task. This process requires a pre-trained model built from a comprehensive set of reference data to solve similar problems in other domains [20].

2.2. CNNs and TL Forms

Large convolutional neural networks (CNNs) are among the most widely used pre-trained models in TL due to their high performance, ease of training and versatility in various computer vision tasks [21]. A typical CNN model consists of two main components: the convolutional basis and the classifier. The convolutional basis, which consists of convolutional layers and pooling layers, acts as a feature extractor [21]. This base can be further subdivided into three sub-extractors: low-level, mid-level and advanced-level feature extractors. The classifier, on the other hand, usually consists of fully concatenated (FC) layers that classify the image based on the extracted features.

Figure 2 illustrates the components of a typical CNN model, focusing on the convolutional basis and the classifier in the context of a white blood cell detection task. The low-level feature extractor captures basic elements such as curves, edges and corners. The mid-level feature extractor identifies features related to the nucleus and cytoplasm of different blood cells. This process culminates in the capture of comprehensive features for complete or partial blood cells, which are then classified by an MLP (multi-layer perceptron) head into categories such as eosinophils, lymphocytes, monocytes or neutrophils.

Figure 3 shows the three different TL forms in CNNs, which represent the variations when replacing classifier and convolution base layers [22–27]:

- First TL form Figure 3a: In this form, the last FC layer of the classifier is replaced. The weights of all other layers are frozen, except for the new FC layer, which is retrained. This method is useful when only a small amount of data is available.
- Second TL form Figure 3b: In this form, the entire classifier is replaced while the weights of the convolutional basis are frozen. The new classifier can be an FC layer or another ML classifier such as SVM or RF. This form requires more data than the first one.
- Third TL form Figure 3c: In this form, both the classifier and part or all of the convolutional basis are replaced. This method requires the most data of the three forms.

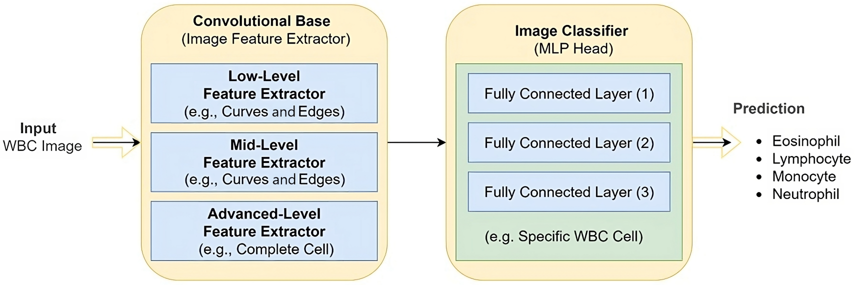


Figure 2. Architecture of a typical CNN model for WBC classification.

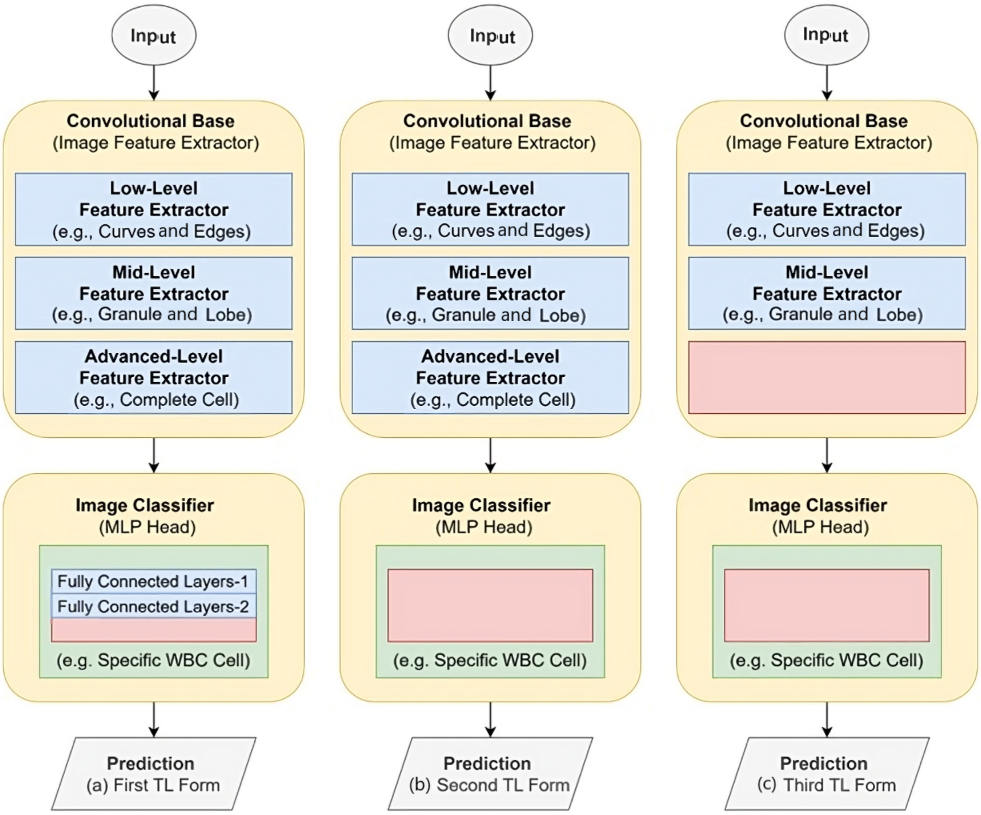


Figure 3. Three forms of transfer learning in CNNs.

2.3. Learning Impacting Factors

This work studies the path of making self-conscious ImageNet models. This requires that these models have the basic tools for continuously measuring, estimating and monitoring their level of knowledge. Such a tool is derived by the analysis of their behavior through transfer learning.

During the TL process, the amount of data represents only one parameter of the knowledge formula for ImageNet models. Further parameters include the image quality of the new dataset and its similarity to the ILSVRC dataset. To this end, two sets of peripheral blood smear data, “PBC” [4] and “BCCD” [5], are selected to analyze the development of the learning process in ImageNet models.

As a general rule, a new dataset is considered small if each class of that dataset contains less than 1000 images per class. Each class of the PBC dataset offers more than 1000 high-quality images. In contrast, every class in the BCCD dataset contains less than 500 low-quality images. Data augmentation techniques (DATs), such as image mirroring, are applied to produce more than 1000 images per class. Thankfully, this DAT application will help the degradation in image quality. This means that we will eventually have two datasets of equal size for the same objects "blood cell images", but with opposing image qualities.

3. Materials and Methods

This section outlines the methodology employed to classify white blood cell (WBC) images into various categories using pre-trained deep neural network models. The datasets used include two peripheral blood smear datasets, namely "PBC" and "BCCD." The detailed methodology is illustrated in Figure 4.

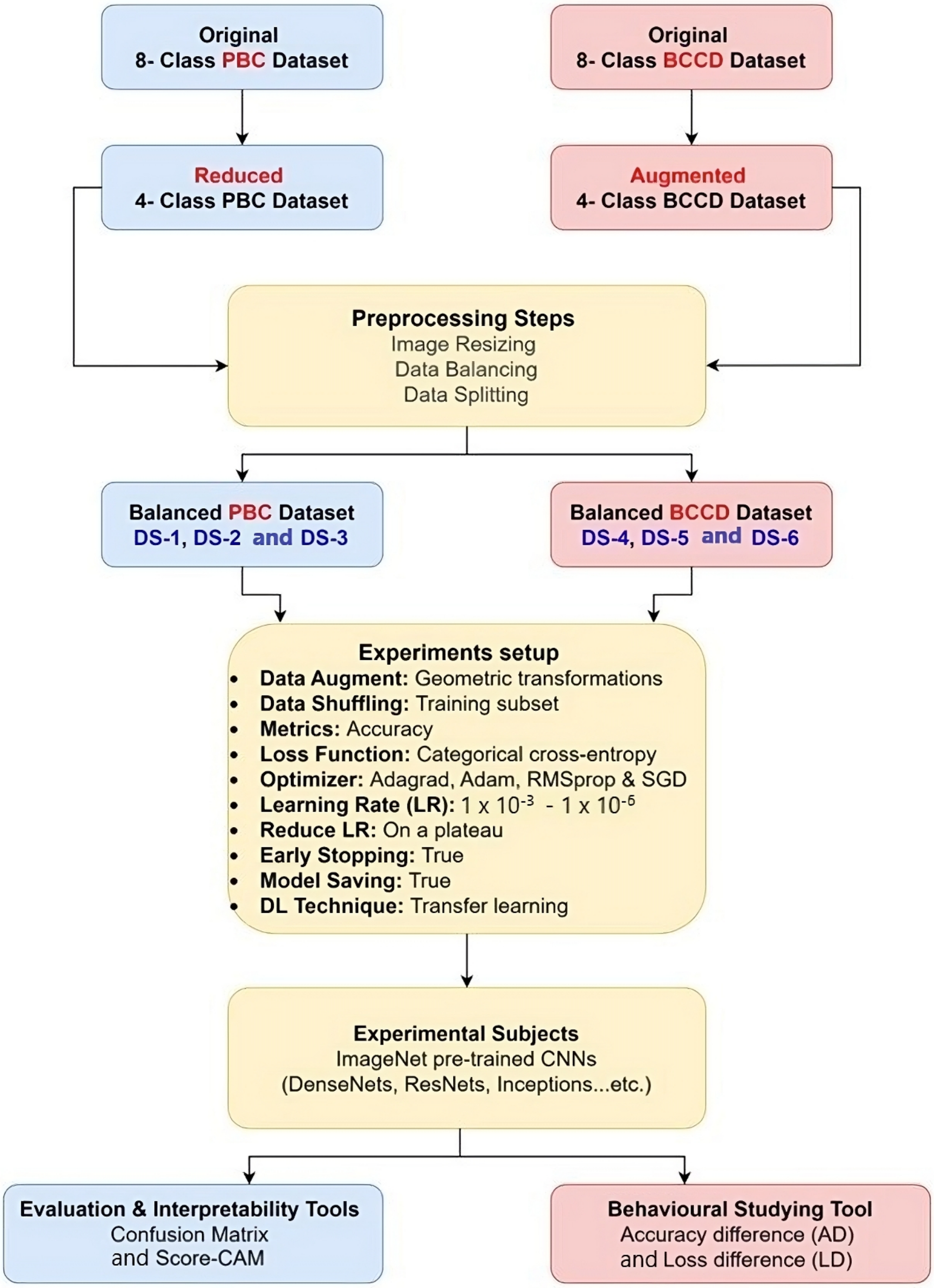


Figure 4. Detailed methodology.

3.1. PBC and BCCD Datasets

The PBC dataset comprises 17,092 images categorized into eight blood cell groups [4], as summarized in Table 1.

Table 1. Summary of the PBC dataset.

#	Cell Type	Total Number of Images per Class	Percentage (%)
1	Neutrophils	3329	19.48
2	Eosinophils	3117	18.24
3	Basophils	1218	7.13
4	Lymphocytes	1214	7.10
5	Monocytes	1420	8.31
6	Immature Granulocytes (IG)	2895	16.94
7	Erythroblasts	1551	9.07
8	Platelets (Thrombocytes)	2348	13.74
	Total	17,092	100

The PBC images have a standard size of 360 × 363 pixels, which is very close to the input requirements of the ImageNet models and thus minimizes the effects of resizing the images. Figure 5 [4] shows the images of each blood cell type in the eight groups.

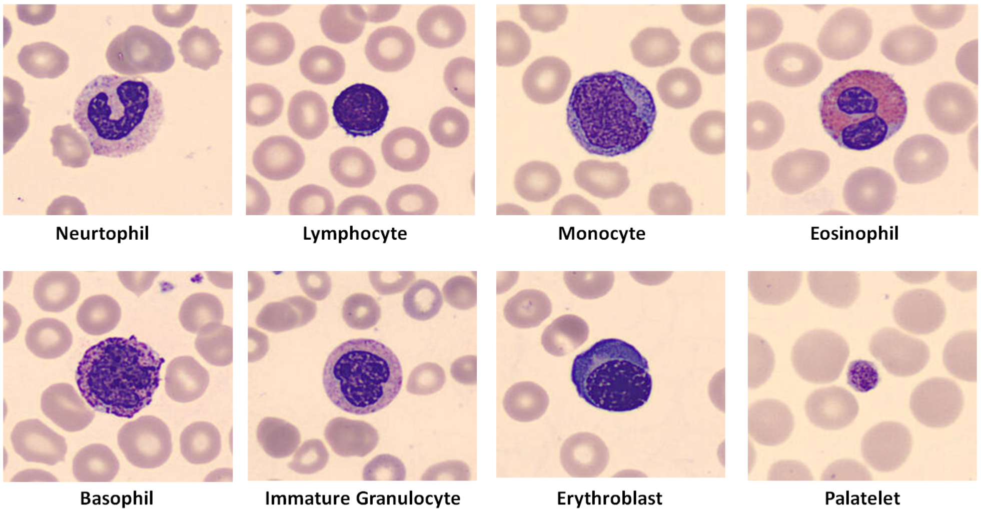


Figure 5. Eight PBC blood cell types.

The BCCD dataset originally comprises 410 peripheral blood smears of red blood cells (RBCs), white blood cells and platelets [5]. The images are in JPEG format and have a size of 640 × 480 pixels. Figure 6 shows typical images for each of the four WBC types in the original BCCD dataset.

These blood smears often contain accumulated erythrocytes and oil droplets, which contributes to the lower image quality in the BCCD dataset. Table 2 summarizes the distribution of eosinophils, lymphocytes, monocytes and neutrophils in the original BCCD dataset [5].

Table 2. WBCs summary in the original BCCD dataset.

#	Cell Type	Total Number of Images per Class	Percentage (%)
1	Neutrophils	88	25.2
2	Eosinophils	207	59.3
3	Lymphocytes	33	9.5
4	Monocytes	22	6.0
	Total	349	100

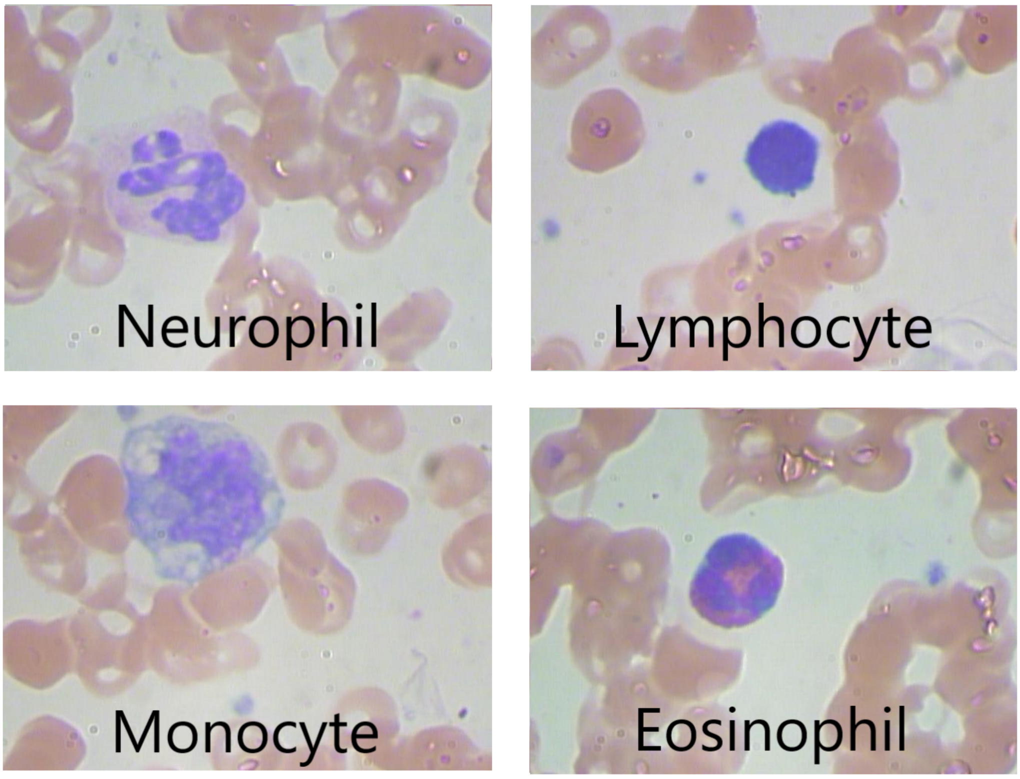


Figure 6. Four BCCD blood cell types.

3.2. Reduced PBC and Augmented BCCD Datasets

Because the BCCD dataset includes only four WBC classes while the PBC dataset contains eight, it is necessary to exclude the additional four WBC classes from the PBC dataset. This adjustment ensures that both datasets have four classes each.

Moreover, because the BCCD dataset has a limited number of images per class, it requires initial augmentation to create a balanced BCCD dataset. The employed augmentation techniques are image transformations such as image rotating and shear, and they are randomly applied to produce enough data.

Table 3 represents a summary of the four WBCs distribution in the new augmented BCCD dataset [28].

Table 3. WBCs summary in the augmented BCCD dataset.

#	Cell Type	Total Number of Images per Class	Percentage (%)
1	Neutrophils	3171	25.3
2	Eosinophils	3133	25.0
3	Lymphocytes	3109	24.8
4	Monocytes	3102	24.8
	Total	12,515	100

Figure 7 shows four sample frames for each of the four WBC types in the new expanded BCCD dataset [28].

As a result, we obtain a reduced 4-class PBC dataset and an augmented 4-class BCCD dataset.

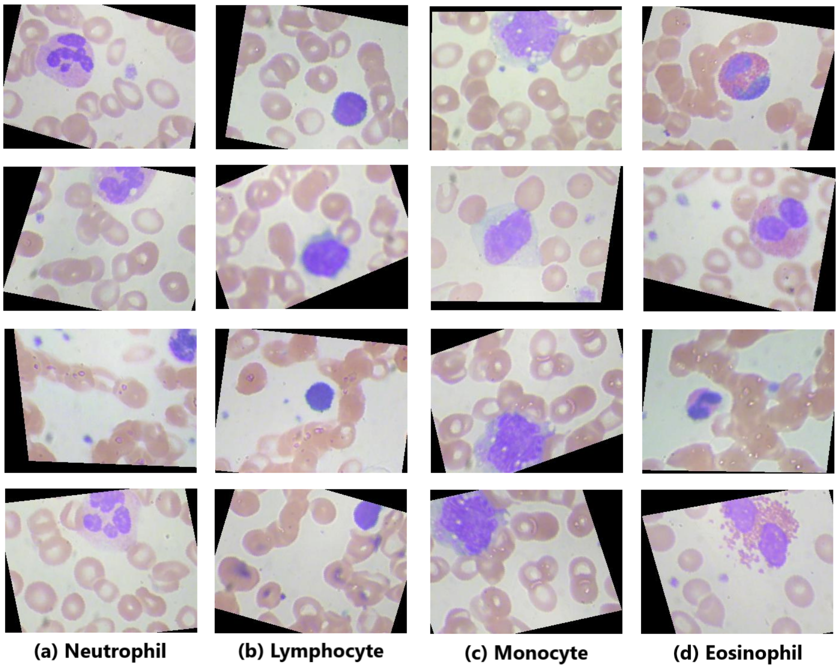


Figure 7. (**a**) Neutrophil: augmented image showing neutrophil cells, subjected to random transformations such as rotation and shear. (**b**) Lymphocyte: augmented image of lymphocyte cells, generated by applying random transformations like rotation and shear. (**c**) Monocyte: Example of monocyte cells with image augmentation, including rotation and shear. (**d**) Eosinophil: augmented eosinophil image created using transformations such as rotation and shear.

3.3. Preprocessing Steps

Dataset pre-processing includes three steps: image resizing, data balancing, and splitting.

First, images in both the PBC and BCCD datasets need to be resized to fit the standard "240 × 240" image input of ImageNet models.

Secondly, the analysis of behavior demands keeping a minimum number of evaluating metrics during comparison. This target is achieved through balanced datasets using accuracy and loss as assessment tools. Three balanced datasets (Table 4), DS-1, DS-2 and DS-3, will represent the PBC dataset. Of the eight blood cell types in the PBC dataset, we have included only four: neutrophils, lymphocytes, monocytes and eosinophils. This selection aims to maintain consistency by targeting the same number of classes in both the newly balanced PBC dataset and the BCCD dataset.

Table 4. PBC-balanced datasets: DS-1, DS-2 and DS-3.

Blood Cell Types	PBC Dataset—DS-1	PBC Dataset—DS-2	PBC Dataset—DS-3
Neutrophils	200	400	1000
Lymphocytes	200	400	1000
Monocytes	200	400	1000
Eosinophils	200	400	1000
Total Number	800	1600	4000
Training	720	1440	3600
Validating	80	160	400

Three balanced WBC datasets (Table 5), DS-4, DS-5 and DS-6, will represent the BCCD dataset.

The final data pre-processing step involves dividing data into training and validating data with a 10-to-1 fold for each new PBC and BCCD datasets.

Table 5. BCCD-balanced datasets: DS-4, DS-5 and DS-6.

Blood Cell Types	**BCCD Dataset—DS-4**	**BCCD Dataset—DS-5**	**BCCD Dataset—DS-6**
Neutrophils	400	1000	2750
Lymphocytes	400	1000	2750
Monocytes	400	1000	2750
Eosinophils	400	1000	2750
Total Number	1600	4000	11,000
Training	1440	3600	9900
Validating	160	400	1100

3.4. Balanced PBC and BCCD Datasets

After the pre-processing steps, we obtain three balanced PBC datasets (DS-1, DS-2 and DS-3) and three balanced BCCD datasets (DS-4, DS-5 and DS-6). This diversity in terms of dataset size and image quality enables an investigation of their effects on the learning behavior of the pre-trained ImageNet CNNs through the transfer learning process.

The next step is to standardize the experimental setup for all common models to provide a solid foundation for subsequent analysis.

3.5. Unified Experimental Setup

The experimental setup is standardized with the following elements: data augmentation techniques, data shuffling, metrics used, employed loss function used, learning rate range, automated learning rate reduction, early stopping, model storage and the deep learning technique used.

Geometric transformations are applied as augmentation techniques to improve model generalization. Data shuffling is performed only on training subsets to ensure better generalization. "Accuracy" is chosen as the metric, and "Categorical Cross-Entropy" is used as the loss function. The optimizers used include AdaGrad [29], ADAM [30], RMSprop [31], and SGD [32].

The learning process involves selecting a range of learning rates (LRs) from 1×10^{-3} to 1×10^{-6}. The reduction of the learning rate reduction is controlled by the "Plateau" function with a certain degree of patience level. The learning process is completed with the "Early Stopping" function. The trained model is saved for later analysis and evaluation.

The deep learning technique used in these experiments is transfer learning (TL), which utilizes the pre-trained experience of the ImageNet CNNs to avoid the need for large datasets, extensive computational resources and time.

3.6. Experimental Subjects: ImageNet Pre-Trained CNNs

The ImageNet competition marked a pivotal moment in the evolution of computer vision by exposing the shortcomings of shallow models with few hidden layers and spurring the development of deep architectures such as ResNet, DenseNet and VGGs. Prior to these advances, shallow machine learning models struggled to effectively update synaptic weights across numerous layers, limiting their ability to learn complex features from data. This limitation made them unsuitable for large-scale image classification tasks.

However, the introduction of deep architectures with innovative techniques such as residual connections, dense connectivity and hierarchical feature extraction overcame these barriers. These advances facilitated the training of deep networks by tackling problems such as vanishing gradients and enabling the effective propagation of information across multiple layers. As a result, the ImageNet competition has triggered the proliferation of deep learning methods that are revolutionizing computer vision and advancing artificial intelligence.

The focus of this study is on analyzing the behavior of the ImageNet models rather than discussing their architectures in detail. The 11 ImageNet models used in this study include Xception [33]; VGG-16 and VGG-19 [34]; ResNet-50, ResNet-101 and ResNet-152 V2 [35]; Inception V3 [36]; InceptionResNet V2 [37]; and DenseNet-121, DenseNet-169 and DenseNet-201 [38].

The VGG models [34] marked a significant milestone in the ImageNet competition by demonstrating the effectiveness of deep architectures in large-scale image classification tasks. The VGG models used a simple but powerful architecture with a stack of convolutional layers followed by max-pooling layers, achieving remarkable success. This unified structure enabled the effective extraction of complicated features at different spatial resolutions and the acquisition of rich representations of the input images.

The straightforward design of the VGG architecture facilitated implementation and interpretation, making it a popular choice for researchers and practitioners. VGG models demonstrated unprecedented accuracy in image classification, underscoring the transformative potential of deep learning methods. VGG's success in the ImageNet competition has underscored the importance of deep architectures and catalyzed further advances in computer vision research.

Figure 8 illustrates the architecture of the VGG-16 model [34] using the example of classifying a neutrophil from the PBC dataset [4]. The VGG-16 model is characterized by its deep structure, which consists of 16 layers and effectively recognizes complicated features within the neutrophil image. Each layer contains convolutional operations followed by max-pooling layers that enable hierarchical feature extraction across multiple scales. This architecture enables the VGG-16 model to capture fine-grained details essential for accurate classification tasks and shows exceptional performance in distinguishing different cell types.

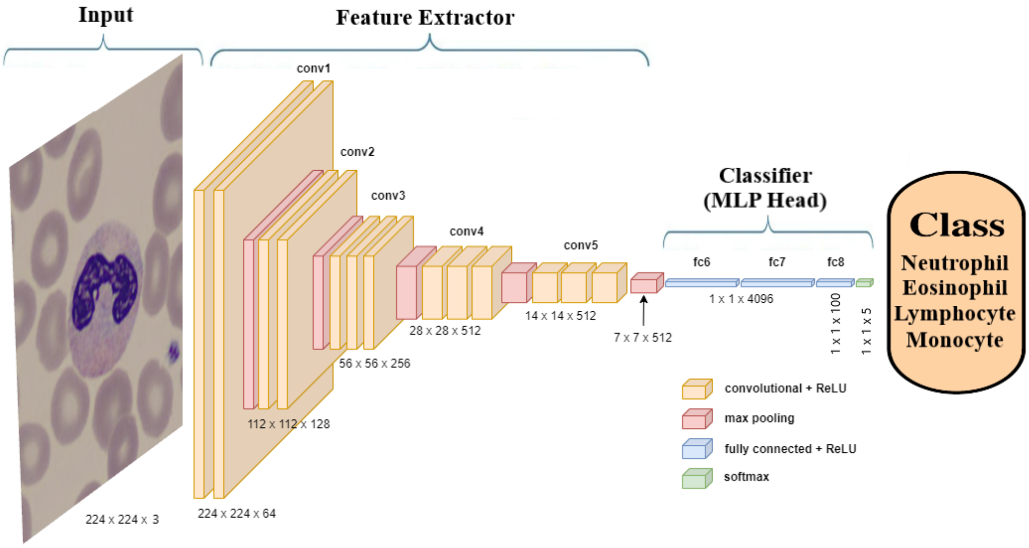


Figure 8. Architecture of the VGG-16 model for the classification of a neutrophil.

ImageNet accuracy is a well-established metric for models trained on the ILSVRC dataset with fast processors over an extended period of time. In contrast, transfer learning (TL) accuracy depends on the specific TL approach, the size of the new dataset, its similarity to ILSVRC and the quality of the images [13]. These factors are usually determined by AI researchers. Novice AI researchers often lack the expertise to accurately assess these factors accurately, which can lead to the overfitting of deep learning models. This emphasizes the importance of incorporating sensors that track the progress of the learning process history to achieve self-awareness.

3.7. Evaluation and Interpretability Tools

The first TL form is used in this study. Different optimizers (SGD, RMSprop, AdaGrad and ADAM) and loss functions (categorical cross-entropy and MSE) were tested, with the best results being obtained with the ADAM optimizer and the categorical cross-entropy loss function. In order to maintain consistency, the results of the 50th epoch were used, and functions such as "TerminateOnNaN", "ReduceLROnPlateau" and "EarlyStopping" were used initially.

Table 6 presents the constant parameters used in the experiments. These include the ADAM optimizer with a default learning rate of 0.001, the categorical cross-entropy loss function, the accuracy metric, a total of 50 epochs and a training-to-validation ratio of 90% to 10%.

Table 6. Experiments—constant parameters and their corresponding values.

	Constant Parameter	Values
1	Optimizer	ADAMs with defaults
2	Loss function	Categorical cross-entropy
3	Metrics	Accuracy
4	Epochs	50
5	Training-to-validation ratio	90%/10%

Google Colaboratory (Google Colab) was used for the experiments. Google Colab is a valuable machine learning and data analytics tool that allows data scientists to write and run Python code via an online-hosted Jupyter notebook [39].

For evaluation and interpretability, confusion matrices with the corresponding classification reports [8] and Score-CAM are used to numerically and visually analyze and evaluate the performance of the models. In addition, two new parameters, accuracy difference (AD) and loss difference (LD), are developed as potential indicators of self-awareness ability.

Confusion matrices are analyzed numerically using classification report parameters such as accuracy, precision, recall and F1-score [8].

Accuracy is the ratio of correctly predicted classes to the total number of samples evaluated [8]. The equation for accuracy is:

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

where *TP* and *TN* stand for true positive and true negative cases, respectively, and *FP* and *FN* stand for false positive and false negative cases, respectively.

Precision measures the quality of a positive prediction of the model [8]. The equation for precision is:

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

Recall measures how many of the true positives were correctly identified by the model [8]. The equation for recall is:

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

The F1-score is the harmonic mean of precision and recall [8]. The equation for the F1-score is:

$$\text{F1} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{4}$$

Score-CAMs are used to visualize the internal focus of feature maps of models [40]. They provide a visual assessment of model performance alongside numerical assessment through classification reports.

3.8. Tool for Behavioral Studies

The core of this work is the search for potential indicators that could provide a concise history of the learning process in pre-trained deep learning models. Typically, researchers compare training and validation accuracy curves and loss curves to assess over-

or underfitting during the learning process. In this study, it is proposed to directly measure the differences between these curves as potential indicators of model behavior.

Accuracy difference (*AD*) is the difference between the training accuracy (*TA*) and the validation accuracy (*VA*). The equation for *AD* is:

$$AD = TA - VA \tag{5}$$

The loss difference (*LD*) is the difference between the training loss (*TL*) and the validation loss (*VL*). The equation for *LD* is

$$LD = TL - VL \tag{6}$$

These differences serve as potential indicators for achieving self-awareness in AI models. They provide insights into the learning process and guide necessary adjustments to improve model performance.

3.9. Theoretical Framework for AD and LD

In traditional machine learning, overfitting and underfitting are evaluated by analyzing the gap between training and validation performance, typically through accuracy and loss curves. AD and LD simplify this process by offering quantitative measures of these gaps at any point during training.

From a mathematical standpoint, AD and LD represent the degree of divergence between the model's internal representation of the data (captured by training accuracy and loss) and its generalization capability (captured by validation accuracy and loss). The larger the AD and LD, the more the model is either overfitting or underfitting, which reflects the model's inability to balance learned knowledge with generalization to unseen data.

Conceptually, AD and LD can be seen as measures of the model's "awareness" of its performance, functioning similarly to self-monitoring mechanisms in conscious entities. This self-awareness emerges when the model is able to recognize the divergence between its training and validation phases, enabling adjustments to its learning process. The use of AD and LD allows the model to "detect" this divergence and signal whether it is overconfident in its learning (overfitting) or underconfident (underfitting).

3.10. Comparison with Existing Methods

While AD and LD are useful for detecting overfitting or underfitting, they should be considered alongside traditional performance metrics like accuracy, precision, recall, and confusion matrices. Unlike early stopping or regularization techniques (e.g., dropout, L2 regularization), which prevent overfitting, AD and LD help monitor learning progression without intervening directly. However, these metrics may have limitations, particularly in highly imbalanced datasets or in noisy conditions, where standard methods such as confusion matrices and cross-validation could provide a more detailed view of model performance.

In certain edge cases, AD and LD may not effectively capture model behavior. For example, in imbalanced datasets where accuracy alone is not an accurate indicator of model performance, AD might misleadingly show progress, even when precision or recall are poor. Similarly, in cases with noisy data, the loss difference (LD) might fluctuate significantly, obscuring the true learning behavior. Therefore, in such situations, incorporating advanced regularization methods and additional metrics may be required to ensure a robust evaluation of model performance.

We now highlight that while learning and validation curves reveal trends in model performance, AD and LD offer a concise, singular metric that quantifies the difference between training and validation results, making them easier to interpret. Furthermore, methods like test time augmentation and prediction confidence scores focus on robustness during inference, whereas AD and LD focus on behavior during training, providing different but complementary perspectives.

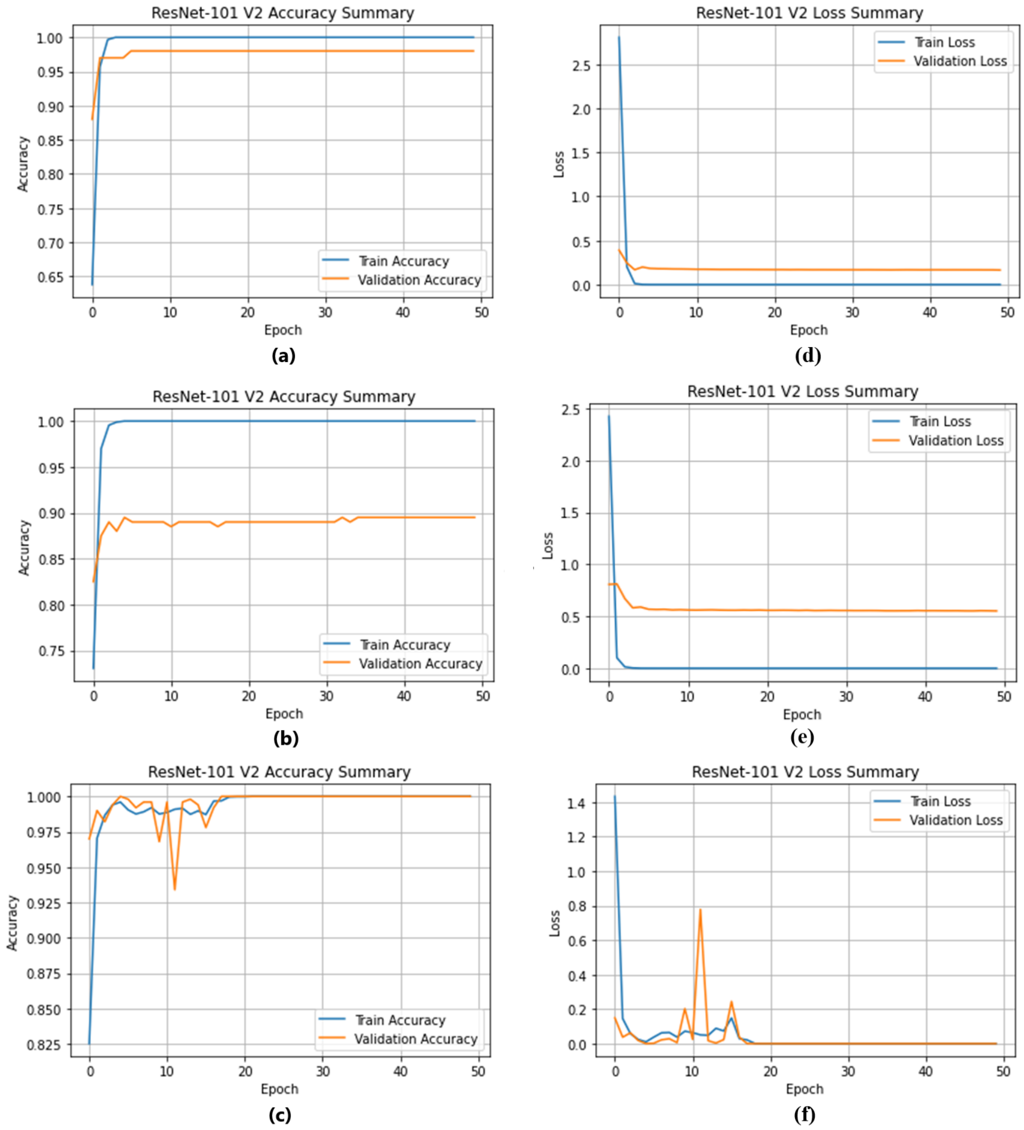
4. Results

This section presents the results of the transfer learning process with eleven ImageNet models on six balanced datasets created from two peripheral blood smear datasets, PBC and BCCD. The balanced datasets from the PBC dataset are DS-1, DS-2 and DS-3, and those from the BCCD dataset are DS-4, DS-5 and DS-6.

Because all 11 ImageNet models showed similar behavior, the results for the ImageNet ResNet-101 V2 model are presented graphically for simplicity, while summary tables contain the numerical results for the other models.

4.1. Balanced PBC Dataset Results: DS-1, DS-2 and DS-3

Figure 9 shows the accuracy summary results of fitting the ResNet-101 V2 model with the PBC-balanced datasets: DS-1, DS-2 and DS-3.



(a)

(d)

(b)

(e)

(c)

(f)

Figure 9. Summary of accuracy and loss results for ResNet-101 V2 on PBC-balanced datasets. The subfigures show: (**a**) Accuracy curve for DS-1, (**b**) Accuracy curve for DS-2, (**c**) Accuracy curve for DS-3, (**d**) Loss curve for DS-1, (**e**) Loss curve for DS-2, and (**f**) Loss curve for DS-3.

Figure 9a,b show that the ResNet-101 V2 model has an average AD value of +2% and +11% (see Table 7) when using the DS-1 and DS-2 datasets, respectively. After fitting with the DS-3 dataset, the AD value fluctuates between negative and positive values (Figure 9c) before finally reaching 100% accuracy (see Table 7).

Figure 9d,e show that the ResNet-101 V2 model has an average LD value of −0.16 and −0.55 when fitted with the DS-1 and DS-2 datasets, respectively. This pattern indicates a significant overfitting of the ResNet-101 model due to the data size. When using the DS-3 dataset (Figure 9f), the LD value alternates between negative and positive values before finally reaching zero loss.

The identification of overfitting is evident in both the DS-1 and DS-2 datasets, as shown in Figure 9. To address this, the introduction of the DS-3 dataset proves to be a crucial solution. This improvement is due to the considerable increase in the number of images, which emphasizes the importance of an adequate size of the dataset to mitigate the problem of overfitting. It is important to recognize that mitigating overfitting through robust dataset size is a nuanced aspect that may be overlooked by some researchers.

Table 7 provides a summary of the average AD and LD values for ImageNet models fitted with the balanced PBC datasets.

Table 7. Summary of Average AD and LD difference Values for ImageNet models fitted with balanced-PBC datasets.

ImageNet Models	**Average AD Value**			**Average LD Value**		
	DS-1 (%)	**DS-2 (%)**	**DS-3 (%)**	**DS-1**	**DS-2**	**DS-3**
DenseNet-121	+4	+5	0	−0.10	−0.20	0
DenseNet-169	+2	+4	0	−0.06	−0.18	0
DenseNet-201	+1	+2	0	−0.01	−0.15	0
Inception V3	+2	+5	0	−0.08	−0.24	0
Inception-ResNet V2	+1	+5	0	−0.02	−0.16	0
ResNet-50 V2	+6	+7	0	−0.30	−0.34	0
ResNet-101 V2	+2	+11	0	−0.16	−0.55	0
ResNet-152 V2	+7	+9.5	0	−0.15	−0.48	0
VGG-16	+2	+7	0	−0.07	−0.21	0
VGG-19	+1	+5	0	−0.08	−0.25	0
Xception	+5	+6	0	−0.16	−0.41	0

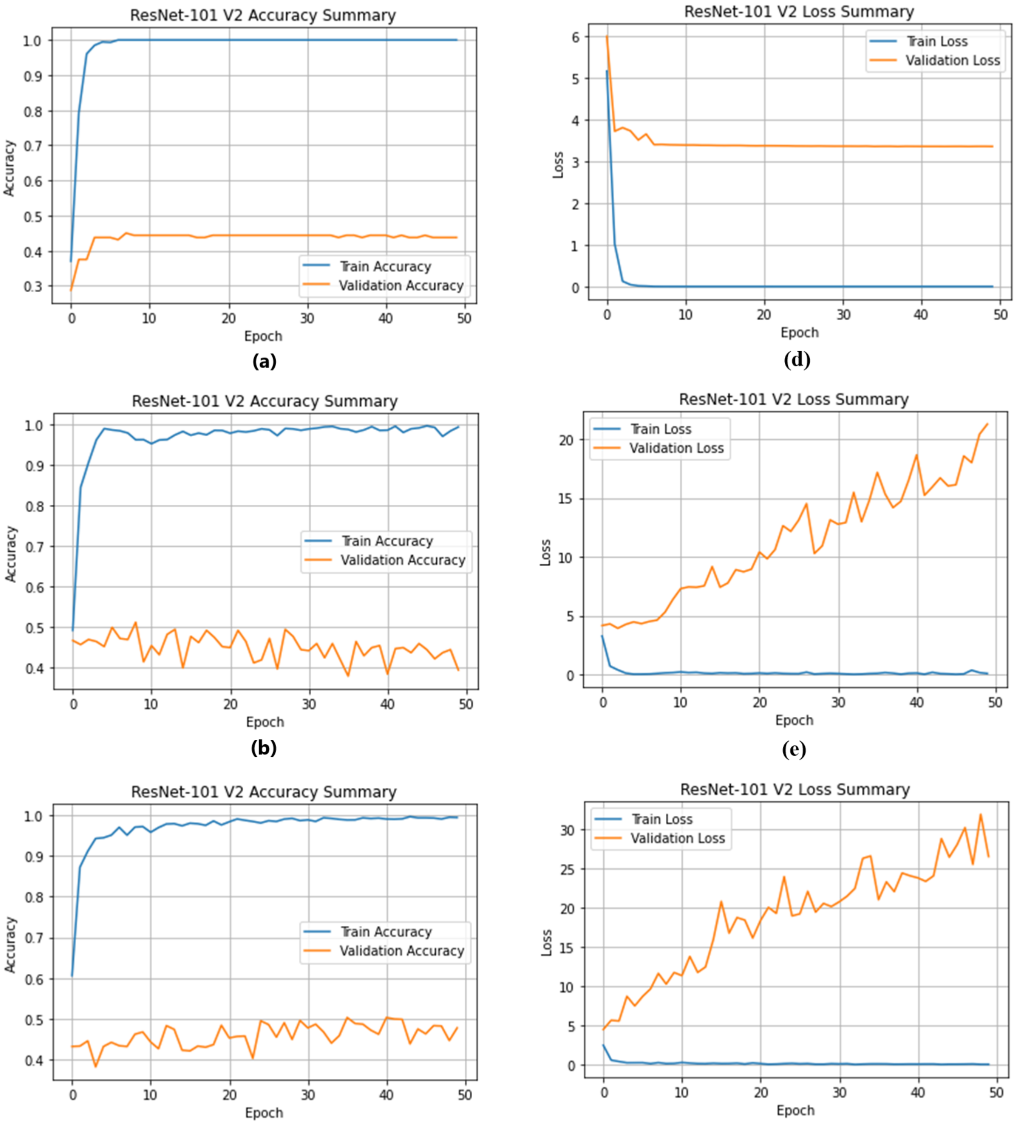
The learning behavior of the other ImageNet models mirrors that of the ResNet-101 V2 model. The only differences lie in the average AD and LD values, which vary slightly and are either slightly higher or lower.

4.2. Balanced BCCD Dataset Results: DS-4, DS-5 and DS-6

Figure 10 shows the accuracy summary results of fitting the ResNet-101 V2 model with the BCCD-balanced datasets: DS-4, DS-5 and DS-6.

Figure 10a–c show that the ResNet-101 V2 reaches an average AD value of +56%, +55% and +52% (see Table 8) with the use of the BCCD-balanced datasets. The DS-4 AD curve has less variation in comparison with the DS-5 and DS-6 AD curves.

Figure 10d–f show that the ResNet-101 V2 reaches an average LD value of −3.35, +15.05 and +20 (see Table 8) with the use of the BCCD-balanced datasets. All DS-4, DS-5 and DS-6 TL training curves are horizontal lines with a near-zero average value. The DS-4 VL curve is nearly a horizontal line with a value of 3.35. However, the DS-5 and DS-6 VL curves raise exponentially with high fluctuations. Such behavior demonstrates clear overfitting, which becomes even worse with increasing data size and consequently noise.



(a) **(d)** **(b)** **(e)** **(c)** **(f)**

Figure 10. Summary of accuracy and loss results for ResNet-101 V2 on BCCD-balanced datasets. The subfigures show: (**a**) Accuracy curve for DS-4, (**b**) Accuracy curve for DS-5, (**c**) Accuracy curve for DS-6, (**d**) Loss curve for DS-4, (**e**) Loss curve for DS-5, and (**f**) Loss curve for DS-6.

Table 8 shows the average AD and LD difference values for ImageNet models fitted with the BCCD-balanced datasets using the first TL form.

Table 8. Average AD and LD values for ImageNet models fitted with the balanced-BCCD datasets.

ImageNet Models	**Average AD Value**			**Average LD Value**		
	DS-4 (%)	**DS-5 (%)**	**DS-6 (%)**	**DS-4**	**DS-5**	**DS-6**
DenseNet-169	+50	+43	+45	−2.00	−7.50	−17.50
DenseNet-201	+48	+46	+42	−2.25	−8.50	−16.50
Inception-ResNet V2	+45	+40	+42	−1.45	−1.65	−8.50
ResNet-50 V2	+59	+55	+55	−3.41	−14.50	−25.00
ResNet-101 V2	+56	+55	+52	−3.35	−15.05	−20.00
ResNet-152 V2	+55	+48	+45	−2.40	−2.80	−15.00
VGG-16	+49	+48	+45	−1.40	−1.50	−2.00

4.3. DenseNet-169 Model Results

This study used the DenseNet-169 model as a representative example to illustrate the classification reports of the ImageNet models. These summaries show the performance of the model when trained on different datasets, including PBC DS-2 (400 images per class), PBC DS-3 (1000 images per class), BCCD DS-4 (400 images per class) and BCCD DS-5 (1000 images per class).

Figure 11a,b show the confusion matrices for the DenseNet-169 model trained with the PBC DS-3 (1000 images per class) and BCCD DS-5 (1000 images per class) datasets, respectively.

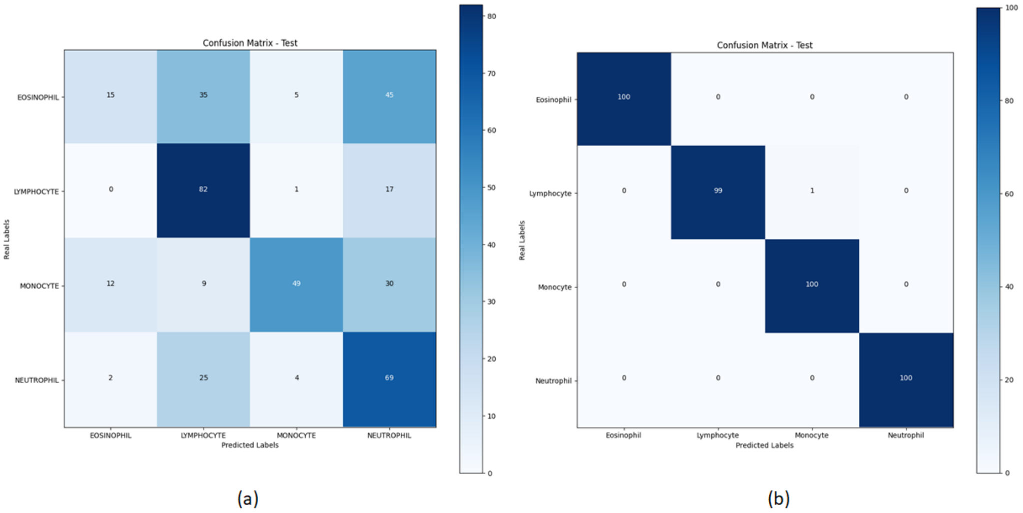


Figure 11. DenseNet-169 model confusion matrix trained with: (**a**) the BCCD DS-5 dataset; (**b**) the PBC DS-3 dataset.

The summary of classification reports for the DenseNet-169 model across different datasets, including PBC DS-2 (400 images per class), PBC DS-3 (1000 images per class), BCCD DS-4 (400 images per class) and BCCD DS-5 (1000 images per class), is shown in Table 9.

Table 9. DenseNet-169—the datasets' quality impact.

Datasets	Macro Average—(Balanced Datasets)			
	Precision	Recall	F1-Score	Accuracy
PBC DS-2	0.98	0.98	0.98	0.98
PBC DS-3	1.00	1.00	1.00	1.00
BCCD DS-4	0.62	0.56	0.55	0.56
BCCD DS-5	0.58	0.54	0.51	0.54

Table 10 showcases the performance metrics of different optimizers on the PBC DS-3 dataset (1000 images per class). Each optimizer is evaluated based on its precision, recall, F1-score and accuracy. ADAM stands out with perfect scores across all metrics, achieving a remarkable precision, recall, F1-score and accuracy of 1.00. Following closely behind is RMSprop, displaying high scores across the board with a precision of 0.98, recall of 0.99, F1-score of 0.98 and accuracy of 0.98. AdaGrad and SGD also demonstrate strong performance, maintaining consistency with scores of 0.97 in precision, recall, F1-score and accuracy. This comparison underscores ADAM's exceptional performance in handling balanced datasets, setting a high standard for precision-driven optimization.

Table 10. DenseNet-169—the optimizer's impact.

Optimizer	Macro Average—(PBC DS-3 Dataset)			
	Precision	Recall	F1-Score	Accuracy
AdaGrad	0.97	0.97	0.97	0.97
ADAM	1.00	1.00	1.00	1.00
RMSprop	0.98	0.99	0.98	0.98
SGD	0.97	0.97	0.97	0.97

SCORE-CAM (spatially class-discriminative relevance class activation mapping) is a method in deep learning that identifies and highlights important areas in an image contributing to a neural network's classification decision. It helps visualize which parts of an image are significant for a specific classification outcome, aiding in model interpretation. Figure 12 shows the Score-CAM of DenseNet-169 trained with the PBC DS-2 (400 images per class) and DS-3 (1000 images per class) datasets.

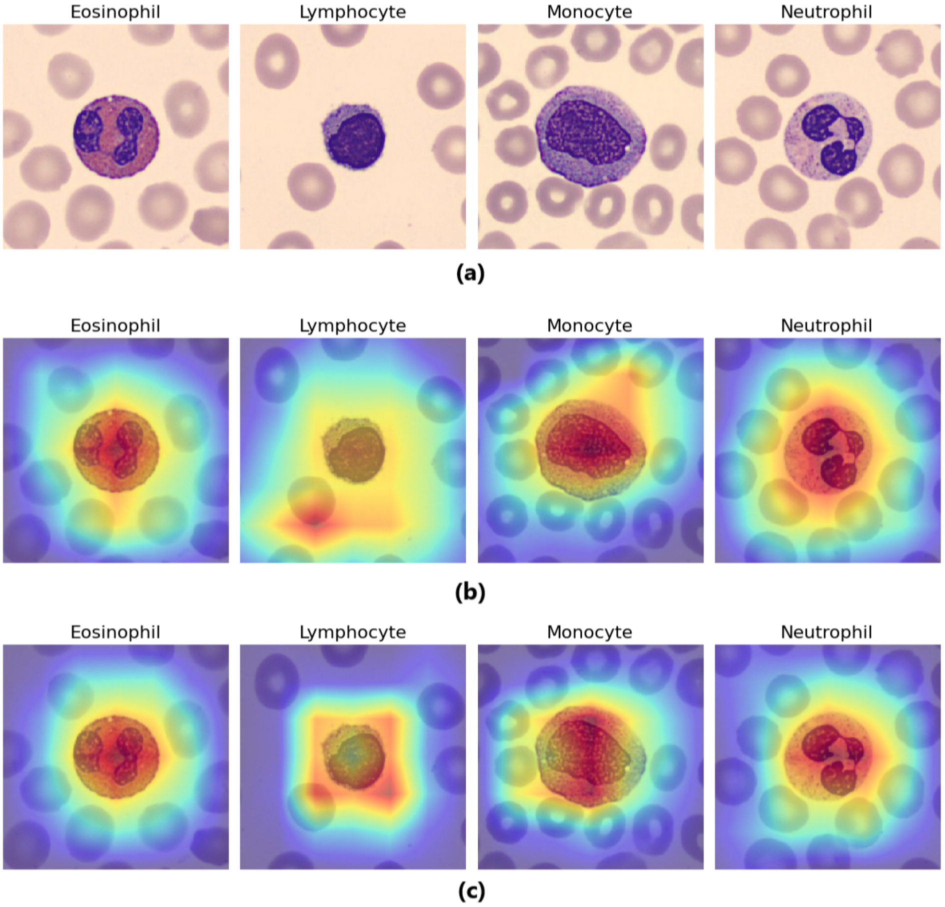


Figure 12. DenseNet-169 model Score-CAM trained with the PBC DS-2 and DS-3 datasets. The subfigures show: (**a**) Testing Images from Original PBC Dataset, (**b**) PBC DS-2 Dataset (400 Images per Class), and (**c**) PBC DS-3 Dataset (1000 Images per Class).

Figure 13 shows the Score-CAM of DenseNet-169 trained with the BCCD DS-4 (400 images per class) and DS-5 (1000 images per class) datasets.

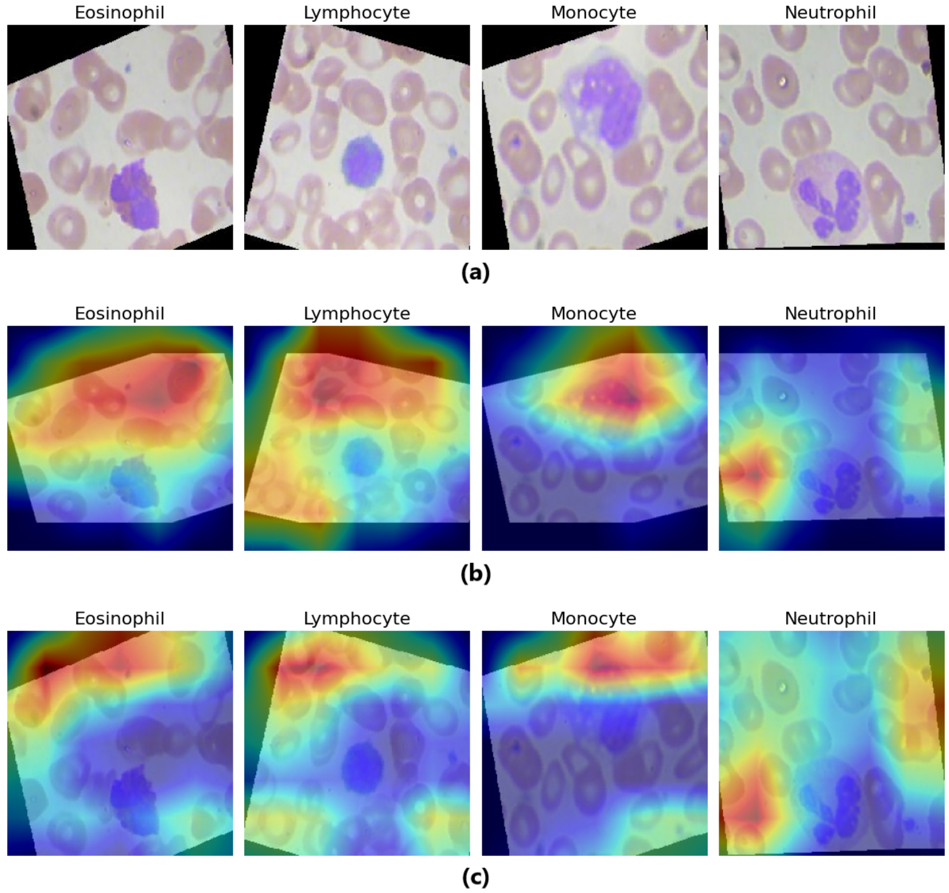


Figure 13. DenseNet-169 model Score-CAM trained with the BCCD DS-4 and DS-5 datasets. The subfigures show: (**a**) Testing Images from Augmented BCCD Dataset, (**b**) BCCD DS-4 Dataset (400 Images per Class), and (**c**) BCCD DS-5 Dataset (1000 Images per Class).

5. Discussion

5.1. Learning Process and Impacting Factors

The results show that ImageNet models exhibit specific learning behavior that is influenced by several factors, including:

- Model: architecture and depth;
- Dataset: size and class representation;
- Image: similarity (feature depth), quality and size;
- Transfer learning (TL) shape: first, second or third.

The effect of model depth is consistent in the two balanced datasets of BCCD and PBC. Increasing the number of CNN layers requires more data, which means that learning behavior can be observed and identified earlier for shallower models.

The use of PBC balanced datasets demonstrates the importance of sufficient data and balanced class representation. In contrast, BCCD balanced datasets show that data augmentation techniques (DATs) are ineffective in compensating for lack of data in small, unbalanced datasets.

One thousand images per class in the PBC DS-3 dataset was sufficient to achieve AD and LD values of zero. However, using 2750 images per WBC class in the BCCD DS-6 dataset resulted in significantly poorer performance.

High-quality images allow models to achieve mature behavior characterized by “zero AD and LD values” with a minimum of data, computational resources and time. Finally, as mentioned earlier, the choice of a particular TL shape can significantly affect the learning progression of ImageNet models.

Our findings align with previous studies on TL in medical image analysis, where dataset size and quality are key factors influencing model performance. However, we observed unexpectedly high overfitting in the BCCD dataset despite augmentation, suggesting that beyond data quantity, intrinsic image quality plays a critical role. This observation is consistent with the recent literature indicating that image quality impacts TL success, but the extent of overfitting observed here, even with large datasets, suggests that further investigation is needed to understand the limits of augmentation techniques in compensating for low-quality data.

Moreover, the contrasting performance between the PBC and BCCD datasets indicates that factors such as dataset quality may be more critical than previously emphasized in the literature. This opens the door for future research into strategies for mitigating overfitting when working with lower-quality datasets, as well as refining augmentation methods to better address issues stemming from data quality.

One limitation of AD and LD is their sensitivity to dataset quality and size. While effective for smaller datasets like PBC and BCCD, these metrics may require adjustment when applied to larger, more diverse datasets. Furthermore, their effectiveness in detecting self-awareness in more complex architectures, such as transformer-based models, remains an open question. Future research will explore how these metrics can be integrated into existing AI systems to improve self-awareness and decision making, particularly in real-time applications like autonomous driving.

5.2. ImageNet Models and Consciousness Path

Accuracy, loss and their respective training and validation difference functions have been shown to be reliable indicators of the learning process in ImageNet models. These parameters serve as effective tools for measuring, monitoring and updating information about the actual experience and maturity of these models. The inclusion of these parameters provides ImageNet models with a kind of self-awareness and gives them a sense of their performance and learning process.

5.3. DenseNet-169 Model Results

The results of the DenseNet-169 model, including the confusion matrices, summary classification reports and score CAMs, contribute significantly to understanding the effectiveness of using AD and LD as censoring elements to build more confident ImageNet models.

In Figures 12 and 13, the focal areas of the Score-CAMs on the testing images from the original PBC dataset and the augmented BCCD dataset illustrate the influence of data quantity in two contrasting scenarios: high-quality and low-quality images. Specifically, Figure 12b,c demonstrate that an increase in the quantity of high-quality images within the PBC dataset enhances the model's ability to focus on the object of interest, namely the four white blood cells. In contrast, Figure 13b,c visually indicate that increasing data quantity in low-quality images, such as those from the BCCD-4 and BCCD-5 datasets, leads to greater model confusion.

6. Conclusions

To summarize, this study represents a pioneering attempt to integrate censorship elements into the development of self-aware artificial intelligence, with a focus on self-aware convolutional neural networks (ConvNets). By examining blood cell classification tasks using the "PBC" and "BCCD" datasets, this research provides invaluable insights into endowing ImageNet models with self-aware capabilities.

The "PBC" dataset, known for its high quality, coherent images and balanced class distribution, contrasts sharply with the challenges of the "BCCD" dataset, which lacks these favorable characteristics. Despite attempts to compensate for the imbalance of the dataset through data augmentation, this study reveals significant challenges, particularly in mitigating accuracy losses, highlighting the complicated interplay between data quality, quantity and model performance.

By analyzing the accuracy difference (AD) and loss difference (LD), this study highlights their central role as censors in investigating the behavior of ConvNets and shedding light on the dynamics of the learning process. These metrics not only serve as indispensable tools for measurement and monitoring, but also pave the way for the introspection of ImageNet models.

While the practical realization of self-aware AI may still seem a long way off, this study serves as a beacon pointing the way for AI researchers to further explore this burgeoning field. By explaining the behavior of ConvNets and highlighting the importance of censorship elements, this work contributes to the foundational knowledge that is essential for the eventual manifestation of self-aware artificial intelligence.

Furthermore, this work lays the foundation for the realization of self-aware AI models and encourages further research to investigate additional censoring elements and various AI tasks beyond image classification. As the field progresses, continued efforts in this direction will undoubtedly pave the way for transformative advances in artificial intelligence and usher in a new era of sentient and self-aware machines.

Future Research Directions

Future research will focus on integrating additional censoring elements to enhance self-awareness, such as uncertainty quantification techniques. Moreover, the exploration of self-aware models in other fields like autonomous systems and robotics will be critical for expanding the applicability of this approach. We also plan to apply explainability tools like SHAP and LIME to further interpret the decision-making process of self-aware AI models, providing more transparency and insight into their internal workings. Additionally, research into addressing the limitations of data augmentation techniques for low-quality datasets will be crucial, especially in medical image analysis, to develop more robust AI systems.

Author Contributions: M.A.A. targeted data curation, investigation and methodology. F.D. addressed conceptualization, investigation and supervision and handled validation and review. I.A.-C. focused on supervision and validation. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partially supported by grant GIU23/022, funded by the University of the Basque Country (UPV/EHU); grant PID2021-126701OB-I00, funded by the Ministerio de Ciencia, Innovación y Universidades, AEI, MCIN/AEI/10.13039/501100011033; and by "ERDF A way of making Europe".

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data used in this paper are publicly available.

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

References

1. Sarker, I.H. Machine learning: Algorithms, real-world applications and research directions. *SN Comput. Sci.* **2021**, *2*, 160. [CrossRef] [PubMed]
2. Cioffi, R.; Travaglioni, M.; Piscitelli, G.; Petrillo, A.; De Felice, F. Artificial intelligence and machine learning applications in smart production: Progress, trends, and directions. *Sustainability* **2020**, *12*, 492. [CrossRef]
3. Xu, Y.; Lu, C.; Zhang, J.; Peng, Z.; Zhou, Y. Artificial intelligence: A powerful paradigm for scientific research. *Innovation* **2021**, *2*, 100179. [CrossRef] [PubMed]
4. 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] [PubMed]
5. Cheng, S. BCCD-Dataset: BCCD (Blood Cell Count and Detection) Dataset Is a Small-Scale Dataset for Blood Cells Detection. Available online: https://github.com/Shenggan/BCCD_Dataset (accessed on 15 April 2024).
6. Shetty, D.; Mehta, K.; Dixit, M.; Chauhan, A. Diving deep into Deep Learning: History, evolution, types and applications. *Int. J. Innov. Technol. Explor. Eng.* **2020**, *9*, 2835–2846. [CrossRef]
7. Janiesch, C.; Zschech, P.; Heinrich, K. Machine learning and deep learning. *Electron. Mark.* **2021**, *31*, 685–695. [CrossRef]
8. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Farhan, L.; Al-Amidie, M.; Santamaría, J. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. [CrossRef]

9. Albawi, S.; Mohammed, T.A.; Al-Zawi, S. Understanding of a convolutional neural network. In Proceedings of the International Conference on Engineering and Technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 1–6. [CrossRef]
10. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
11. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
12. Ribani, R.; Marengoni, M. A survey of transfer learning for convolutional neural networks. In Proceedings of the 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T), Rio de Janeiro, Brazil, 28–31 October 2019; pp. 47–57. [CrossRef]
13. Kornblith, S.; Shlens, J.; Le, Q.V. Do better ImageNet models transfer better? In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 2656–2666. [CrossRef]
14. Signorelli, C.M. Can computers become conscious and overcome humans? *Front. Robot. AI* **2018**, *5*, 121. [CrossRef]
15. Krauss, P.; Maier, A. Will we ever have conscious machines? *Front. Comput. Neurosci.* **2020**, *14*, 556544. [CrossRef]
16. Kim, Y.; Lee, H.J.; Shim, J. Developing data-conscious deep learning models for product classification. *Appl. Sci.* **2021**, *11*, 5694. [CrossRef]
17. Pepperell, A.R. Does machine understanding require consciousness? *Front. Syst. Neurosci.* **2022**, *16*, 788486. [CrossRef]
18. Puigcerver, J.; Montaner, J.; Benavent, J.; Cazorla, M. Scalable transfer learning with expert models. *arXiv* **2020**, arXiv:2009.13239. [CrossRef]
19. Kim, H.E.; Cosa-Linan, A.; Santhanam, N.; Jannesari, M.; Maros, M.E.; Ganslandt, T. Transfer learning for medical image classification: A literature review. *BMC Med. Imaging* **2022**, *22*, 69. [CrossRef] [PubMed]
20. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 9. [CrossRef]
21. Sharma, S.; Sharma, S.; Athaiya, A. Activation functions in neural networks. *Towar. Data Sci.* **2018**, *1*, 310–316. [CrossRef]
22. Abou Baker, N.; Zengeler, N.; Handmann, U. A Transfer Learning Evaluation of Deep Neural Networks for Image Classification. *Mach. Learn. Knowl. Extr.* **2022**, *4*, 22–41. [CrossRef]
23. Dabrowski, M.; Michalik, T. How Effective Is Transfer Learning Method for Image Classification? In Proceedings of the Conference on Computer Science and Information Systems, Warsaw, Poland, 18–20 September 2017; pp. 3–9. [CrossRef]
24. Desai, C. Image Classification Using Transfer Learning and Deep Learning. *Int. J. Eng. Comput. Sci.* **2021**, *10*, 25394–25398. [CrossRef]
25. Oquab, M.; Bottou, L.; Laptev, I.; Sivic, J. Learning and Transferring Mid-Level Image Representations Using Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1717–1724. [CrossRef]
26. Jha, R.; Bhattacharjee, V.; Mustafi, A. Transfer Learning with Feature Extraction Modules for Improved Classifier Performance on Medical Image Data. *Sci. Program.* **2022**, *2022*, 4983174. [CrossRef]
27. Matsoukas, C.; Haslum, J.F.; Sorkhei, M.; Soderberg, M.; Smith, K. What Makes Transfer Learning Work for Medical Images: Feature Reuse and Other Factors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 19–24 June 2022; pp. 9215–9224. [CrossRef]
28. YOLOv7 BCCD. Available online: https://www.kaggle.com/code/muffin3101/yolov7-bccd (accessed on 22 August 2024).
29. Lydia, A.; Francis, S. Adagrad-An Optimizer for Stochastic Gradient Descent. *J. Comput. Inf. Sci.* **2019**, *6*, 566–568.
30. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2017**, arXiv:1412.6980. [CrossRef]
31. Ruder, S. An Overview of Gradient Descent Optimization Algorithms. *arXiv* **2016**, arXiv:1609.04747.
32. Kiefer, J.; Wolfowitz, J. Stochastic Estimation of the Maximum of a Regression Function. *Ann. Math. Stat.* **1952**, *23*, 462–466. [CrossRef]
33. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1800–1807. [CrossRef]
34. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556. [CrossRef]
35. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity Mappings in Deep Residual Networks. *arXiv* **2016**, arXiv:1603.05027. [CrossRef]
36. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2818–2826. [CrossRef]
37. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 4278–4284. [CrossRef]
38. Huang, Z.G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [CrossRef]

39. Bisong, E. Google Cloud Machine Learning Engine (Cloud MLE). In *Building Machine Learning and Deep Learning Models on Google Cloud Platform, A Comprehensive Guide for Beginners*; Apress: Berkeley, CA, USA, 2019; pp. 59–64. [CrossRef]
40. Wang, H.; Wang, Z.; Du, M.; Yang, F.; Zhang, Z.; Ding, S.; Mardziel, P.; Hu, X. Score-CAM: Score-Weighted Visual Explanations for Convolutional Neural Networks. *arXiv* **2020**, arXiv:1910.01279. [CrossRef]

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.