ORIGINAL RESEARCH

Convolutional neural networks based focal loss for class imbalance problem: a case study of canine red blood cells morphology classification

Kitsuchart Pasupa[1] · Supawit Vatathanavaro[1] · Suchat Tungjitnob[1]

Abstract

Morphologies of red blood cells are normally interpreted by a pathologist. It is time-consuming and laborious. Furthermore, a misclassified red blood cell morphology will lead to false disease diagnosis and improper treatment. Thus, a decent pathologist must truly be an expert in classifying red blood cell morphology. In the past decade, many approaches have been proposed for classifying human red blood cell morphology. However, those approaches have not addressed the class imbalance problem in classification. A class imbalance problem—a problem where the numbers of samples in classes are very different—is one of the problems that can lead to a biased model towards the majority class. Due to the rarity of every type of abnormal blood cell morphology, the data from the collection process are usually imbalanced. In this study, we aimed to solve this problem specifically for classification of dog red blood cell morphology by using a Convolutional Neural Network (CNN)—a well-known deep learning technique—in conjunction with a focal loss function, adept at handling class imbalance problem. The proposed technique was conducted on a well-designed framework: two different CNNs were used to verify the effectiveness of the focal loss function and the optimal hyperparameters were determined by fivefold cross-validation. The experimental results show that both CNNs models augmented with the focal loss function achieved higher F_1-scores, compared to the models augmented with a conventional cross-entropy loss function that does not address class imbalance problem. In other words, the focal loss function truly enabled the CNNs models to be less biased towards the majority class than the cross-entropy did in the classification task of imbalanced dog red blood cell data.

Keywords Imbalanced data · Red blood cell classification · Erythrocytes classification · Medical data · Deep learning

1 Introduction

Human red blood cell morphology provides useful information for disease diagnosis. In the same vein, dog red blood cell morphology can give clues about a dog's health to veterinarians. There are five important features of red blood cell morphology classification: (1) shape, (2) size, (3) colour, (4) inclusion, and (5) arrangement. A morphology of red blood cells must be classified accurately in order for a veterinarian to apply an appropriate treatment (Ford 2013). Normally, a pathologist is needed to classify red blood cell morphologies by looking at the cells under a microscope. This standard method requires great expertise in manual classification. It is a very time-consuming qualitative and quantitative process that is prone to error (Tomari et al. 2014). Recently, computer vision and machine learning have been applied to human blood cell classification problem (Tomari et al. 2014; Taherisadr et al. 2013; Habibzadeh et al. 2011; Lee and Chen 2014; Chy and Rahaman 2019; Ross et al. 2006).

In the image classification field, conventional machine learning techniques require humans to manually extract useful features, i.e., converting raw data including image colour pixel values, image shape, and image texture into appropriate representations or feature vectors for a classifier to accurately classify the input image. Nonetheless, there is a

✉ Kitsuchart Pasupa
  kitsuchart@it.kmitl.ac.th

  Supawit Vatathanavaro
  62606061@kmitl.ac.th

  Suchat Tungjitnob
  62606062@kmitl.ac.th

[1] Faculty of Information Technology, King Mongkut's Institute of Technology, Ladkrabang, Bangkok 10520, Thailand

method called representation learning that allows a machine to discover features from raw data automatically.

Deep learning methods are representation-learning methods with multiple levels of representation layers that outperform conventional machine learning techniques in many kinds of tasks including image classification task (Razzak et al. 2018; LeCun et al. 2015). Recently, several studies have employed Convolutional Neural Network (CNN) to tackle blood cell classification problem (Xu et al. 2017; Razzak and Naz 2017; Durant et al. 2017; Zhao et al. 2017; Tiwari et al. 2018; Qin et al. 2018; Rehman et al. 2018). The results of those studies show that deep learning method is efficient in both blood cell feature extraction and classification. Most of those works focused on human blood cells. It has been known that the proportion of normal red blood cell samples found in the majority of people to abnormal red blood cell samples found in patients (the minority) is high. Moreover, the proportion of normal red blood cell samples to abnormal red blood cell samples is also high among patients themselves. To sum up, the rarity of every kind of abnormal red blood cell morphology causes the collected cell morphology data to be imbalanced (Razzak et al. 2018), which is a common problem in real-world datasets (Hospedales et al. 2011; Weiss 2004; Rahman and Davis 2013). This leads deep learning to achieve high prediction accuracy for the majority class and poor prediction accuracy for the minority class (Huang et al. 2016). In this classification task, though, detecting rare classes (minority class) is often more important (Hospedales et al. 2011). To the best of our knowledge, based on the following pieces of literature (Xu et al. 2017; Razzak and Naz 2017; Durant et al. 2017; Zhao et al. 2017; Tiwari et al. 2018; Qin et al. 2018; Rehman et al. 2018), there has been no research concerning class imbalance problem in classification of human blood cells.

This imbalanced data problem has been reported to reduce the performance of some classifiers (Japkowicz and Stephen 2002). Most existing learning algorithms did not perform well for minority class (Rahman and Davis 2013). For over two decades, many class imbalance learning techniques have been developed (Krawczyk 2016). Class imbalance learning can be divided into two main groups, namely, data-level and algorithm-level (Zhang et al. 2018). Common approaches in data-level group are resampling approaches (Chawla et al. 2002; García and Herrera 2009; Garcı et al. 2012; Han et al. 2005; Lim et al. 2016; He et al. 2008; Mazurowski et al. 2008) that aim to balance the training data. Re-sampling techniques can be further divided into three groups depending on the balancing class distribution method (Haixiang et al. 2017):

- Over-sampling method: this method aims to increase minority class samples. Two widely used methods are randomly duplicating minority samples and generating synthetic minority samples.
- Under-sampling method: this method discards some majority class samples in the dataset. The simplest method, random under-sampling, randomly removes majority class samples.
- Hybrid method: this method is a combination of over-sampling and under-sampling methods.

A typical algorithm-level method is a cost-sensitive learning method (Zhang et al. 2018; Zhou and Liu 2006; Zong et al. 2013; Castro and Braga 2013; Datta and Das 2015). It assigns a higher misclassification cost on the minority class (Zhang et al. 2018), making the classifier focus more on the minority class. Lately, there have been many approaches that implement new loss functions for deep imbalanced learning (Lin et al. 2020; Ma et al. 2018; Yue 2017; Sudre et al. 2017; Wang et al. 2016; Zhang et al. 2017; Suresh et al. 2008; Baloch et al. 2019). Unfortunately, there are some disadvantages to those approaches. In re-sampling approaches, over-sampling methods are computationally complex because of an increasing number of samples, while under-sampling methods may end up discarding important information (Baloch et al. 2019). A cost-sensitive learning method is more computationally efficient, but it is difficult to assign an appropriate cost for each class (Haixiang et al. 2017). Most loss function implementations consider only binary classification problems. In this work, we addressed the problem at algorithm-level or loss function because of its lower computational complexity. In the past decade, numerous loss functions have been developed. We selected focal loss function in this study (Lin et al. 2020). It was originally designed to handle highly-imbalanced classes in object detection tasks. This loss function has widely used in many kinds of tasks (Ma et al. 2018; Doi and Iwasaki 2018; Tian et al. 2018; Abulnaga and Rubin 2018) but not in a medical classification task.

Our contributions are as follows:

- We tackled dog red blood cell morphology classification problem by using two well-known deep CNN architectures, e.g., Residual Network (ResNet) (He et al. 2016) and Densely Connected Convolutional Networks (DenseNet) (Huang et al. 2017).
- We applied a focal loss function for a multi-classification task to handle the class imbalance problem in deep CNNs and compared the performance of the focal loss function to that of the cross-entropy loss function, a conventional deep CNN loss function.
- The proposed method was implemented in a well-designed framework for training deep CNN models and for hyperparameter tuning for class imbalance learning.

2 Related work

2.1 Blood cell classification methods

Many researchers have developed methods for automated red blood cell count and classification from images by using several image processing, machine learning, pattern recognition, and computer vision techniques. Habibzadeh et al. (2011) proposed an automated blood cell count system that uses image processing and pattern recognition techniques on histopathological images of red and white blood cells. They applied an image processing algorithm for cell segmentation and then differentiated between red blood cells and white blood cells by their size estimates. Their framework achieved 90% accuracy in red blood cell count. Taherisadr et al. (2013) proposed a red blood cell classification method based on digital image processing. Several features related to shape, internal central pallor configuration of red blood cells, their circularity, and elongation were extracted and utilised in their proposed rule-based system. Lee and Chen (2014) proposed a hybrid neural network classifier to separate between normal and abnormal red blood cells based on their shape and texture features. The accuracy of the proposed method in classifying normal and abnormal was 88.25%. Moreover, they attempted to classify four types of disease—Burr cell, Sickle cell, Horn cell, and Elliptocyte cell—and achieved 91% accuracy. Chy and Rahaman (2019) compared three machine learning algorithms—k-Nearest Neighbour (kNN), Support Vector Machine (SVM), and Extreme Learning Machine (ELM)—in sickle cell anaemia detection. Firstly, they employed image processing techniques including gray-scale image conversion, noise filtering, image enhancement, and morphological operations then fuzzy C means clustering to separate between normal and sickle cells. In that paper, they mainly extracted the following features: (1) geometrical feature, namely, metric value and elongation and (2) statistical features such as mean, standard deviation, variance, skewness, and kurtosis. The results show that ELM classifier was superior to kNN and SVM.

Recently, deep CNN methods have been applied to tackle red blood cell classification task. Razzak and Naz (2017) presented an efficient deep contour aware segmentation approach based on a fully conventional network. They used CNN to extract features from each segmented cell for an ELM classifier. The accuracy on red blood cell classification was 94.71% and 98.68% on white blood cell classification. The accuracy on red blood cell segmentation was 98.12% and 98.16% on white blood cell segmentation. Durant et al. (2017) aimed to evaluate the performance of CNN in red blood cell morphology classification task. They employed some image augmentation techniques to increase the number of training images and then trained three DenseNet models with the same training set but with different random seed initialisers to evaluate the reproducibility and calculate the ensemble predictions. They concluded that DenseNet is suitable for red blood cell morphology classification. Zhao et al. (2017) proposed an automatic detection and classification system for white blood cells from peripheral blood images. They first used different values of red and blue colours (R-B image) to separate white blood cells from red blood cells and then converted R-B image to binary image with a threshold value, making the nucleus more apparent, followed by a morphology operation to remove slight noise. Subsequently, a granularity feature (pairwise rotation invariant co-occurrence local binary pattern or PRICoLBP feature) was extracted to feed into the SVM to separate two classes—eosinophils and basophils—from one another. Consequently, CNN was applied to the other classes to extract features. The deep-learned features were inputted into a random forest classifier to classify the other three types of white blood cells: neutrophil, monocyte and lymphocyte. The proposed method achieved 92.8% classification accuracy. Tiwari et al. (2018) developed a deep learning model to handle a blood cell classification problem. They applied data augmentation to each white blood cell class, from 400 images to 3000 images, to enlarge the training data. Furthermore, they compared their proposed model with Naïve Bayes and SVM. The results show that their proposed model outperformed the traditional machine learning models.

2.2 Class imbalance learning

For the class imbalance problem, many research studies have mainly concentrated on two levels: data-level and algorithm-level (Garcia et al. 2007).

For the data level, the class imbalance problem was manipulated by either over-sampling the minority class or under-sampling the majority class. The general problem of the over-sampling method was that it only replicated the minority class samples randomly, leading to an over-fitting problem (Chawla et al. 2002; Garcia et al. 2007). In order to solve the over-fitting problem, Chawla et al. (2002) proposed a method called Synthetic Minority Over-sampling Technique (SMOTE) that over-sampling the minority class by generating new data points. Lately, Han et al. (2005) introduced a Borderline-SMOTE algorithm, a modified form of SMOTE algorithm. This algorithm over-samples only the minority samples that are close to the class borderline. Another method called data augmentation has been used to over-sample the minority samples (Qin et al. 2018). That paper presented a deep residual learning method for finer white blood cell classification. They applied data augmentation to balance the image samples among classes, e.g., flipped the images, randomly cropped the original

Fig. 1 The effect of a hyperparameter on focal loss
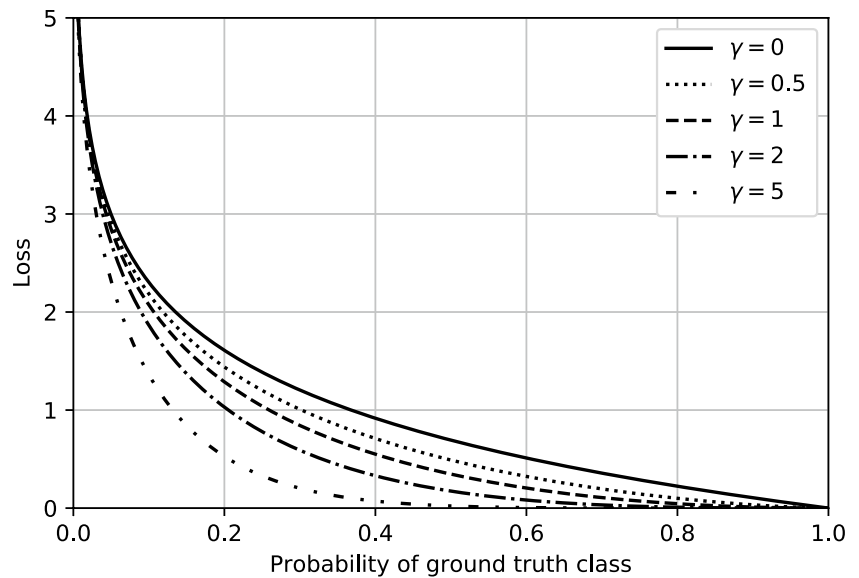
images, added random noise to the original images, etc. However, increasing the training data led to a higher computational cost. On the other hand, under-sampling is preferred as Drummond et al. (2003) demonstrated that under-sampling methods are more efficient than over-sampling methods (Baloch et al. 2019; Garcia et al. 2007).

For algorithm level, an algorithm that emphasises the minority class was employed. A common approach for this kind of level is cost-sensitive learning. Most cost functions give equal importance to each class (Garcia et al. 2007). Therefore, a proper weight needs to be specified for each class. Zong et al. (2013) applied cost-sensitive learning with ELM. They presented weighted ELM for imbalance learning. Their experimental results show that the weighted ELM performed better than the unweighted ELM. Zhang et al. (2018) aimed to tackle class imbalance problem by weighting the cost of each class in Deep Belief Network with an Evolutionary Algorithm. Their proposed method performed significantly better than the others on 58 benchmark datasets and a real-world dataset. Another approach is to modify the loss function in order to make the classifier more sensitive toward minority classes. Lin et al. (2020) proposed a focal loss function to tackle the extreme foreground-background class imbalance for dense object detection by modifying the conventional cross-entropy loss. They demonstrated that a one-stage detector RetinaNet using focal loss function yielded a better performance than those of faster RCNN models. Ma et al. (2018) proposed a semi-focal loss function, modified from a focal loss function, to handle erratic labelling problem in Mitosis Detection. Moreover, they proposed a new mitosis detection network called Cascaded Neural Network with Hard Example Mining and Semi-focal Loss. Their method achieved the best F_1-score at 0.68 in Tumor Proliferation Assessment Challenge 2016.

3 Methodology

3.1 Cross-entropy loss

Cross-entropy loss function is generally used in a deep learning classification model. It is defined as:

$$CE(p, y) = CE(p_t) = -\log(p_t), \tag{1}$$

where

$$p_t = \begin{cases} p, & \text{if } y = 1; \\ 1 - p & \text{otherwise}, \end{cases} \tag{2}$$

p is the prediction probability of the model, and y is ground truth-label.
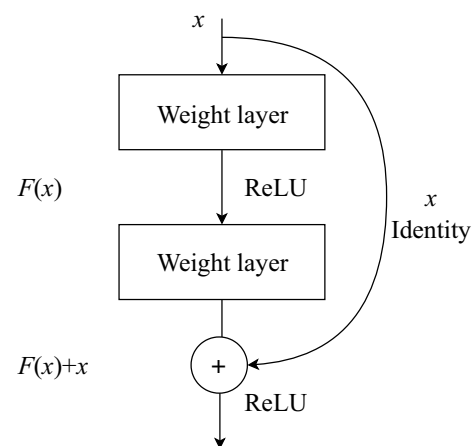

Fig. 2 Residual block

Fig. 3 ResNet-50 architecture

Input Image

Conv $K = 7 \times 7$, $N = 64$, Stride = 2

Max Pool $K = 3 \times 3$, Stride = 2

Conv $K = 1 \times 1$, $N = 64$

Conv $K = 3 \times 3$, $N = 64$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 1 \times 1$, $N = 64$

Conv $K = 3 \times 3$, $N = 64$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 1 \times 1$, $N = 64$

Conv $K = 3 \times 3$, $N = 64$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 1 \times 1$, $N = 128$, Stride = 2

Conv $K = 3 \times 3$, $N = 128$

Conv $K = 1 \times 1$, $N = 512$

Conv $K = 1 \times 1$, $N = 128$

Conv $K = 3 \times 3$, $N = 128$

Conv $K = 1 \times 1$, $N = 512$

Conv $K = 1 \times 1$, $N = 128$

Conv $K = 3 \times 3$, $N = 128$

Conv $K = 1 \times 1$, $N = 512$

Conv $K = 1 \times 1$, $N = 128$

Conv $K = 3 \times 3$, $N = 128$

Conv $K = 1 \times 1$, $N = 512$

Conv $K = 1 \times 1$, $N = 256$, Stride = 2

Conv $K = 3 \times 3$, $N = 256$

Conv $K = 1 \times 1$, $N = 1024$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 3 \times 3$, $N = 256$

Conv $K = 1 \times 1$, $N = 1024$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 3 \times 3$, $N = 256$

Conv $K = 1 \times 1$, $N = 1024$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 3 \times 3$, $N = 256$

Conv $K = 1 \times 1$, $N = 1024$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 3 \times 3$, $N = 256$

Conv $K = 1 \times 1$, $N = 1024$

Conv $K = 1 \times 1$, $N = 256$

Conv $K = 3 \times 3$, $N = 256$

Conv $K = 1 \times 1$, $N = 1024$

Conv $K = 1 \times 1$, $N = 512$, Stride = 2

Conv $K = 3 \times 3$, $N = 512$

Conv $K = 1 \times 1$, $N = 2048$

Conv $K = 1 \times 1$, $N = 512$

Conv $K = 3 \times 3$, $N = 512$

Conv $K = 1 \times 1$, $N = 2048$

Conv $K = 1 \times 1$, $N = 512$

Conv $K = 3 \times 3$, $N = 512$

Conv $K = 1 \times 1$, $N = 2048$

Average Pooling

Fully Connected Layer 1000

Convolution Layer (Conv), Kernel Size (K), Feature Maps (N)

Fig. 4 DenseNet-121 architecture

(a) DenseNet with four dense blocks

(b) Dense Block 1 with 6 layers

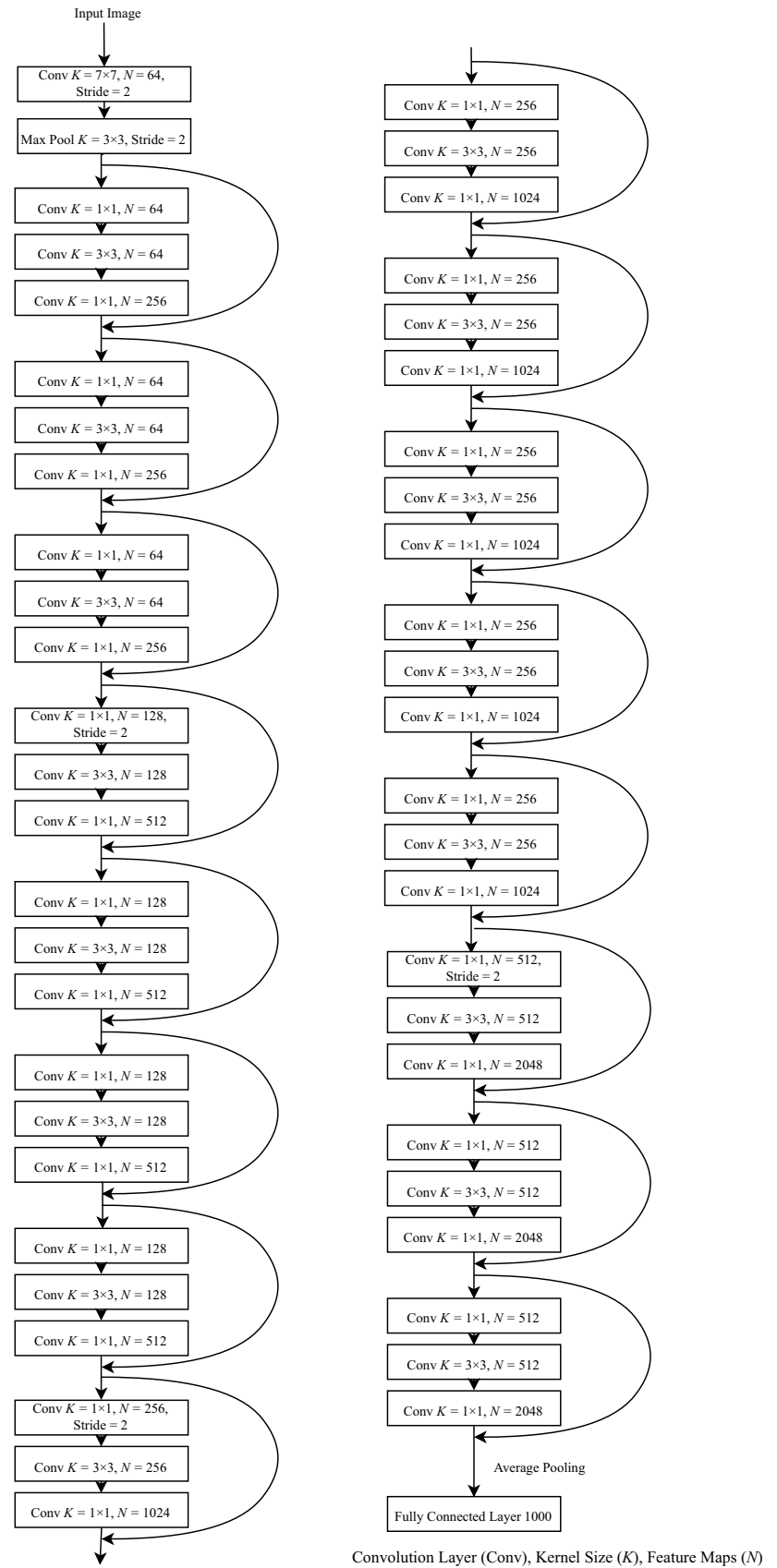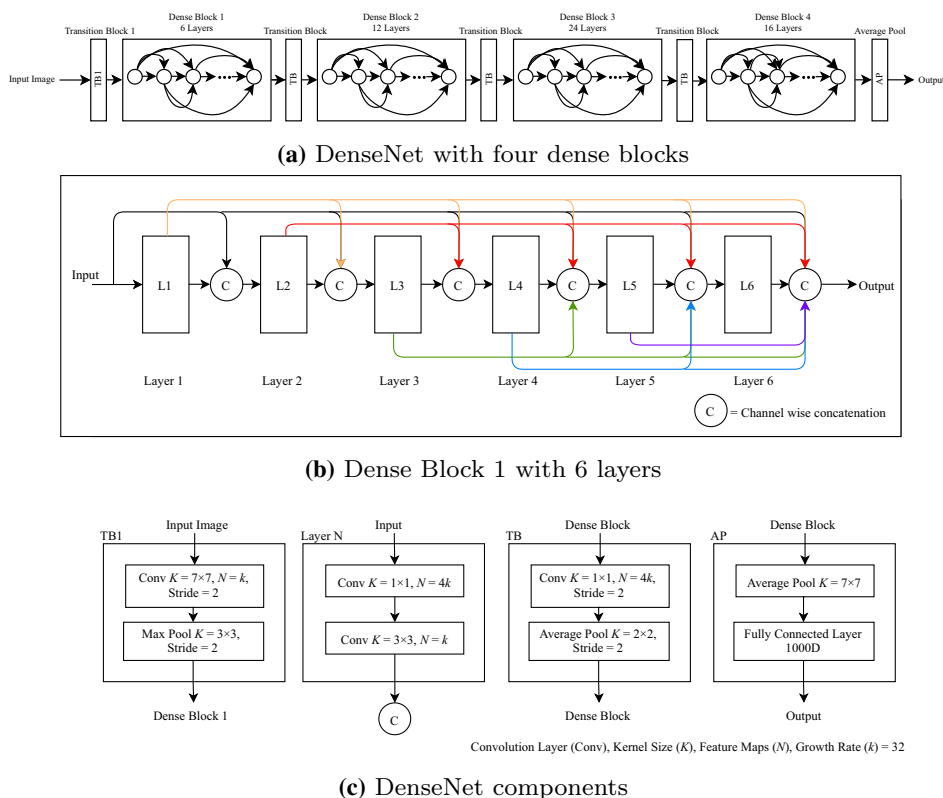Convolution Layer (Conv), Kernel Size (K), Feature Maps (N), Growth Rate (k) = 32

(c) DenseNet components

3.2 Focal loss

Focal loss is modified from cross-entropy loss by adding a modulating factor $(1 - p_t)^\gamma$ to the cross-entropy loss (Lin et al. 2020). It is defined as

$$FL(p_t) = -(1 - p_t)^\gamma \log(p_t). \tag{3}$$

In practice, a focal loss function uses an α-balanced variant of focal loss:

$$FL(p_t) = -\alpha(1 - p_t)^\gamma \log(p_t). \tag{4}$$

γ is a focusing parameter that reshapes the loss function to down-weight easy samples and makes the model focus on hard samples. Hard samples are those samples that produce large errors; a model misclassifies the samples with a high probability. Figure 1 shows the effect of a hyperparameter on focal loss. When $\gamma = 0$, the function behaves like *CE*—presented as a solid line in the figure.

For cross-entropy loss, when a model classifies easy samples—samples in the majority class—to a correct class with $p_t \geq 0.5$, the loss value is low. Although the loss is low, when it is summed over a large number of easy samples, these loss values may overwhelm the rare class in an imbalance data scenario (Lin et al. 2020). This can lead to a biased model towards the majority class.

In focal loss, in the case that the ground truth label is 1 and a sample is correctly classified with a high probability, the value of $(1 - p_t)$ is small. When this term is raised to the power of γ, the value of the modulating factor gets smaller and causes the loss from cross-entropy to be smaller. In contrast, if the model misclassifies a sample with low probability, the modulating factor is large, close to 1; therefore, the loss from cross-entropy remains the same.

3.3 Models

In this research, we aimed to tackle the class imbalance problem in red blood cell morphology classification by using CNN in conjunction with a focal loss function. Many deep CNN architectures have been developed in the past decade. We selected ResNet (He et al. 2016) and DenseNet (Huang et al. 2017) models because both architectures have already been evaluated on ImageNet (Russakovsky et al. 2015) consisting of 1000 classes and gave outstanding results. Moreover, we applied a focal loss function to tackle the highly imbalanced data and compared the performances of both models between using a focal loss function and using a cross-entropy loss function, common in an image classification task.

Fig. 5 Dog blood smear images
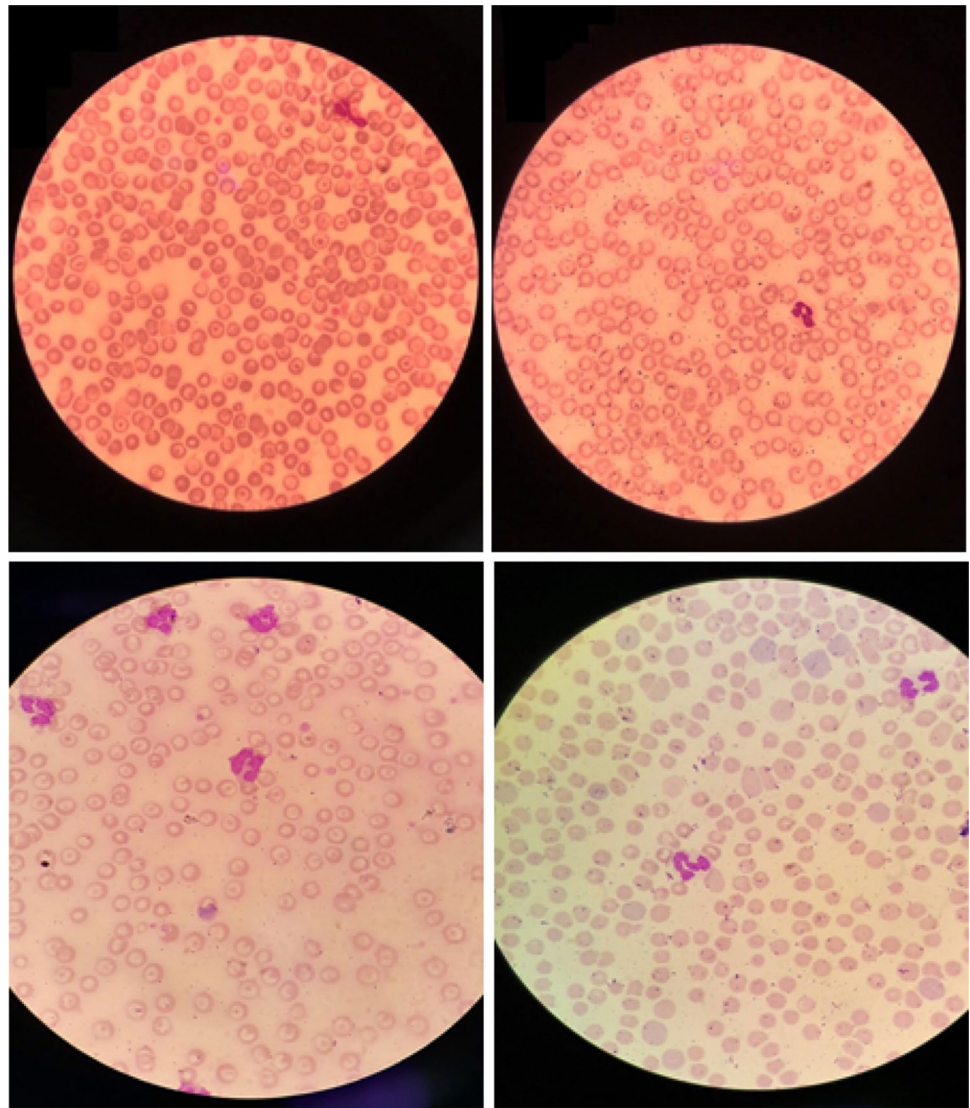
3.3.1 ResNet

It has been known that when a plain network is deep, the back-propagation gradient is small or vanishing, resulting in higher training and test errors (He et al. 2016; Huang et al. 2017; Srivastava et al. 2015; Glorot and Bengio 2010; He and Sun 2015). To solve this vanishing gradient problem, He et al. (2016) proposed a ResNet model. ResNet architecture has a residual block that preserves the gradient. This is done by adding the input x to the output after a few weight layers as shown in Fig. 2, enabling the model to pass useful knowledge from a previous layer to the next. Therefore, this enables the model to have less training error as the network is getting deeper. Furthermore, a ResNet model converges faster compared to a plain network. The experimental results show that ResNet-50, ResNet-101, and ResNet-152 performed better than VGG-16, Goog-LeNet (Inception-v1), and PReLU-Net in top-1 error
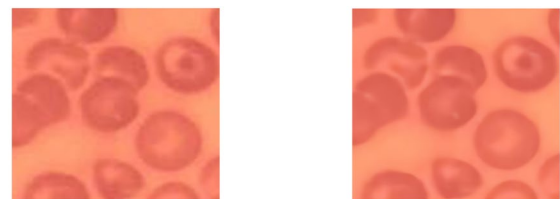
(a) Conventional up-scale method   **(b)** Waifu2×

Fig. 6 Comparison between conventional up-scale method and Waifu2×

and top-5 error on ImageNet (He et al. 2016). Here, we selected ResNet-50, that is 50 layers deep, for our experiment because it is the smallest model. The architecture is shown in Fig. 3.

3.3.2 DenseNet

DenseNet utilises dense connections, in which each layer receives extra inputs from all preceding layers, and also pass its own feature-map to all subsequent layers as shown in Fig. 4. In ResNet, features are combined through summation before passing into a layer. On the other hand, DenseNet combines features by concatenating them to all subsequent layers. Each layer gains the collective knowledge of all other layers, resulting in a thinner and compact network. Moreover, DenseNet clearly achieved a higher accuracy with a smaller number of parameters than ResNet did (Huang et al. 2017). We employed DenseNet-121 in this research because it is the smallest available architecture. The model is shown in Fig. 4.

4 Experimental framework

4.1 Dataset

The 22 dog blood smear images were taken by a camera through a microscope at 100× magnification. The images were provided by the Veterinary Teaching Hospital, Kasetsart University, Hua Hin, Thailand. Examples of dog blood smear images are shown in Fig. 5.

Since the default size of the input image for both ResNet-50 and DenseNet-121 was 224×224 pixels, we fed, one at a time, a single blood cell image with this size into the models. Therefore, we normalised the blood smear images to a size that each blood cell matches 224×224 pixels. Therefore, the input images were required to be enlarged by a scaling factor. A scaling factor was calculated by dividing 224 with the diameter of the red blood cell. If we simply applied the conventional up-scaling method, each image would not be clear and would be full of noise (as shown in Fig. 6a). Then, we employed an image super-resolution tool called "Waifu2×" which was based on the CNNs (Dong et al. 2015) to up-scale all images. This technique enhanced the overall resolution of the output images as shown in Fig. 6b.

After the image normalisation process, adaptive histogram equalisation algorithm was applied to increase the contrast between the background and the red blood cells as shown in Fig. 7b. Then, we employed a Hough circle transform algorithm to detect the red blood cells in a smear image as shown in Fig. 7c. Next, we segmented all red blood cells (as shown in Fig. 7d). These images were then transferred to a pathologist at Kasetsart University to label all the segmented cells. It should be noted that the segmented image size was close to 224×224. We added zero-padding to all segmented red blood cell images to make them 224×224 in size as shown in Fig. 7e. We describe the data pre-processing procedure in Algorithm 1. All red blood cells were labelled and distinguished into three main groups by the pathologist. Our dataset contained 3392 cell images divided into three classes: (1) 345 Codocyte cells (Target cells), (2) 356 Hypochromia cells, and (3) 2691 Normal cells. Examples of red blood cell morphology images are shown in Fig. 8.

Algorithm 1 Cell segmentation algorithm

1: **for all** Smear images **do**
2:      UpscaleRatio $\leftarrow \frac{\text{ExpectedCellDiameter}}{\text{CellDiameter}}$
3:      UpscaledImage $\leftarrow$ Waifu2×(SmearImage, UpscaleRatio)
4:      NormalisedImage $\leftarrow$ Normalise(UpscaledImage)   ▷ Normalise to greyscale image
5:      AdaptImage $\leftarrow$ AdaptiveHistrogramEqualisation(NormalisedImage)   ▷ Increase contrast between background and cells
6:      [CellsCoordinates, Radius] $\leftarrow$ HoughCircleTransform(AdaptImage)
7:      **for all** {CellsCoordinates, Radius} **do**
8:          CellImage $\leftarrow$ SegmentCell(CellCoordinates, Radius)
9:          **if** $2 \cdot \text{Radius} \leq \text{ExpectedCellDiameter}$ **then**
10:             PaddingSize $\leftarrow$ ExpectCellDiameter - (2 · Radius)
11:             PaddedImage $\leftarrow$ ZeroPadding(CellImage, PaddingSize)
12:             CellImage $\leftarrow$ PaddedImage
13:          **end if**
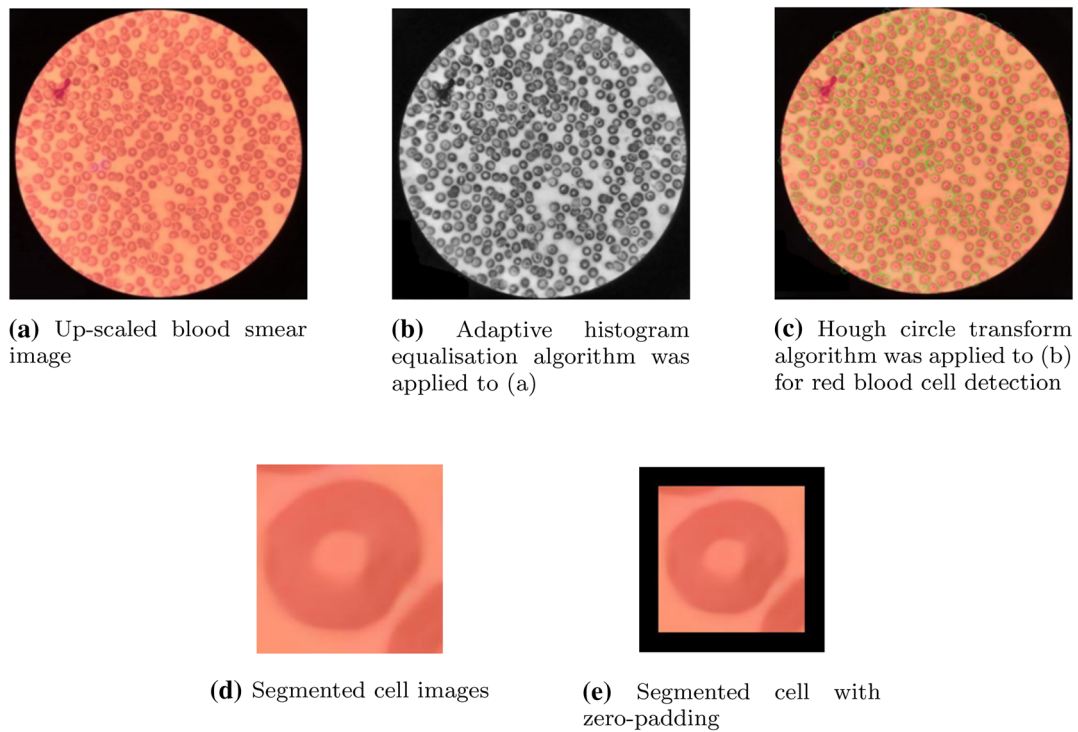14:   SaveImage(CellImage)
15:      **end for**
16: **end for**

(a) Up-scaled blood smear image

(b) Adaptive histogram equalisation algorithm was applied to (a)

(c) Hough circle transform algorithm was applied to (b) for red blood cell detection

(d) Segmented cell images

(e) Segmented cell with zero-padding

Fig. 7 Cell segmentation process

(a) Codocyte (Target Cell)

(b) Hypochromia

(c) Normal
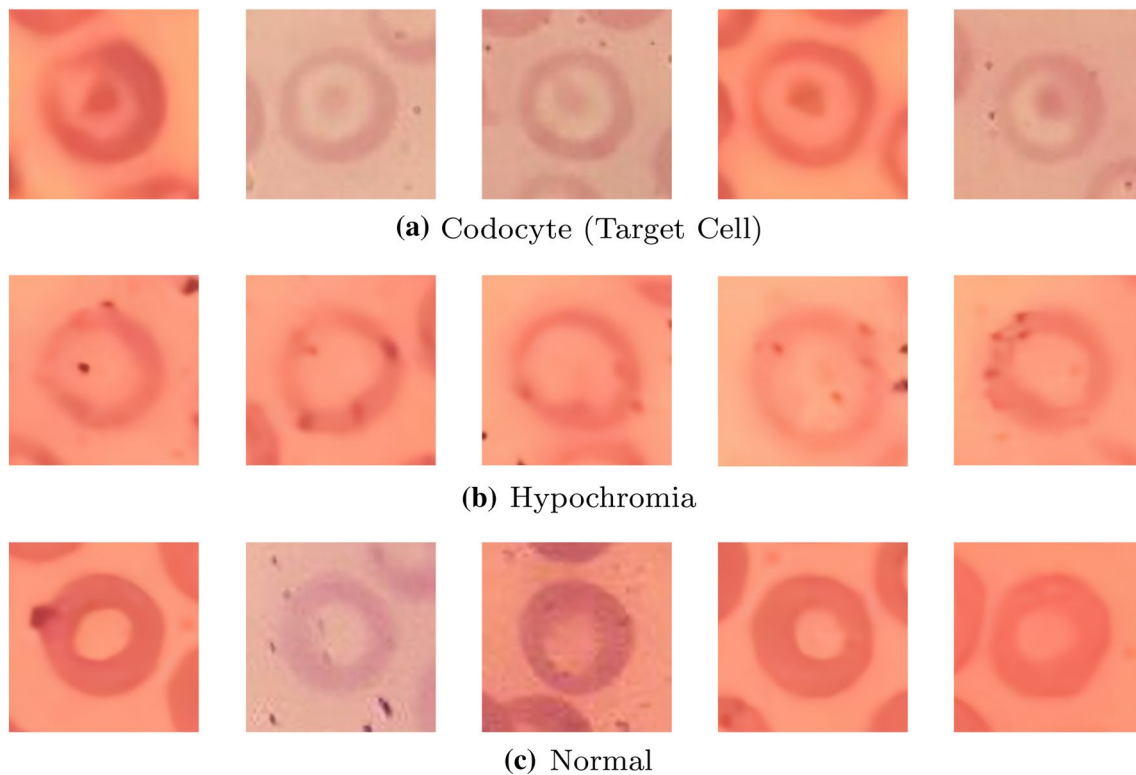
Fig. 8 Three classes of red blood cell morphology in this experiment
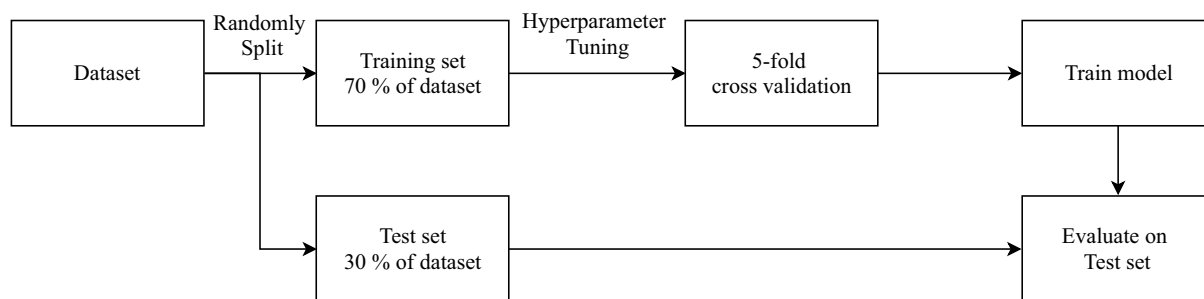
Fig. 9 Process diagram

Table 1 Accuracy, F_1-score, and AUROC achieved by the models with the optimal epoch for each hyperparameter α in the focal loss and cross-entropy loss functions

Model architecture	Method	Hyperparameters		Optimal epoch	Accuracy	F_1-score	AUROC
		α	γ		(%)		
DenseNet-121	Focal loss	0.25	2	89	91.94	0.87	0.98
		0.50		68	82.81	0.72	0.96
		0.75		98	88.80	0.78	0.96
		1.00		76	92.24	0.85	0.98
		1.25		76	92.93	0.86	0.98
		1.50		88	**95.60**	**0.92**	**0.99**
		1.75		64	83.01	0.55	0.95
	Cross-entropy loss	–	–	75	91.30	0.84	0.95
	Under-sampling	–	–	93	88.70	0.80	0.97
	Over-sampling	–	–	95	95.28	0.91	0.99
ResNet-50	Focal loss	0.25	2	34	83.50	0.61	0.93
		0.50		83	86.44	0.71	0.96
		0.75		36	89.49	0.81	0.94
		1.00		90	92.80	**0.87**	0.97
		1.25		79	92.04	0.85	0.97
		1.50		93	89.19	0.83	0.97
		1.75		73	85.17	0.78	0.93
	Cross-entropy loss	–	–	57	91.70	0.85	0.96
	Under-sampling	–	–	68	84.68	0.78	0.94
	Over-sampling	–	–	75	**92.93**	**0.87**	**0.98**

The best performances are expressed in bold text

4.2 Experiment settings

We first randomly split 70% of our dataset as a training set and the remaining 30% as a test set. Since there are a number of hyperparameters to be tuned, we utilised five-fold cross-validation to evaluate the settings of the models on the training set. In the focal loss function, there are two hyperparameters, namely, α and γ as shown in (4). In this experiment, we simply set $\gamma = 2$ because it worked the best in (Lin et al. 2020). Therefore, we concern only on α that varies in the range of [0.25, 0.50, 0.75, 1.00, 1.25, 1.50,

1.75]. The cross-entropy function had no hyperparameter. However, there was still a model hyperparameter—the number of epochs—to be acquired. The number of epochs was determined based on the minimum value of the corresponding loss. Apart from this hyperparameter, we set the input batch size to 32 and the maximum number of epochs to 100. Adam method was utilised as the optimiser (Kingma and Ba 2014), and the learning rate was set to 0.001 for both models. After we obtained the optimal set of parameters, we trained the models on the training set and evaluated the models on the test set as shown in Fig. 9.
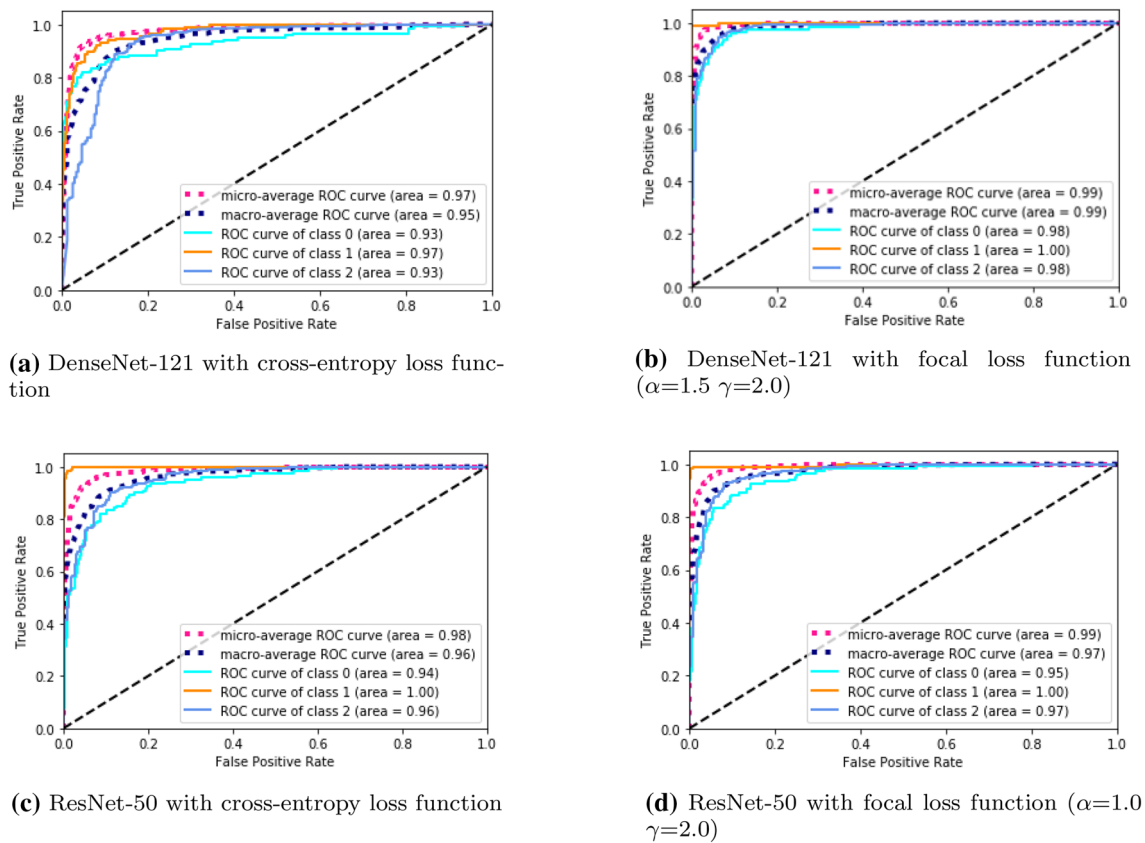
(a) DenseNet-121 with cross-entropy loss function

(b) DenseNet-121 with focal loss function (α=1.5 γ=2.0)

(c) ResNet-50 with cross-entropy loss function

(d) ResNet-50 with focal loss function (α=1.0 γ=2.0)

Fig. 10 Comparison of AUROC achieved by models with cross-entropy loss function and models with focal loss function. (class 0 is Codocyte (Target Cell), class 1 is Hypochromia, and class 2 is Normal)

We compared the models with focal loss function against the models with cross-entropy function. In addition, we employed re-sampling techniques including over-sampling and under-sampling techniques in our task. For the under-sampling technique, we randomly discarded the majority class samples until the number of samples in the majority class was equal to the number of samples in the minority class. For the over-sampling technique, we employed an image augmentation technique by randomly rotating the image between -15 and $15°$ and randomly flipping the image left, right, up, and down to increase the number of samples in the minority class to be equal to the number of samples in the majority class.

It is known that accuracy is commonly used for model evaluation. However, under this imbalanced data scenario, a model could achieve high accuracy by predicting only the majority class. Therefore, we also reported the Area Under the Receiver Operating Characteristic Curve (AUROC) and F_1-score. These have been frequently used as evaluation measures for imbalanced data problems (Haixiang et al. 2017).

5 Results and discussion

We first investigated the effect of the focal loss function hyperparameter on the performances of the two models. It should be noted that the γ hyperparameter of the focal loss function was fixed to 2 because it was a reported best value (Lin et al. 2020). We varied α from 0.25 to 1.75 with 0.25 step size and reported accuracy, F_1-scores and AUROC, shown in Table 1. The F_1-Score achieved by both models with focal loss function slightly increased as α increased, until it reached a peak at $\alpha = 1.5$ for DenseNet-121 and $\alpha = 1.0$ for ResNet-50. DenseNet-121 with focal loss function outperformed the models with cross-entropy loss function in 4/7 cases, while Resnet-50 with focal loss function did so in 2/7 cases. We further compared the performances of ResNet-50 and DenseNet-121 with focal loss function to the two models with conventional cross-entropy loss function in combination with re-sampling techniques. All functions were run with their optimal hyperparameters. Shown in Table 1, DenseNet-121 model with focal loss function achieved the best performance with 0.92 F_1-score. It can also be seen that employing focal loss function in both models

Confusion Matrix

(a) DenseNet-121 with cross-entropy loss function

Confusion Matrix

(b) DenseNet-121 with focal loss function ($\alpha=1.5$ $\gamma=2.0$)

Confusion Matrix

(c) ResNet-50 with cross-entropy function

Confusion Matrix

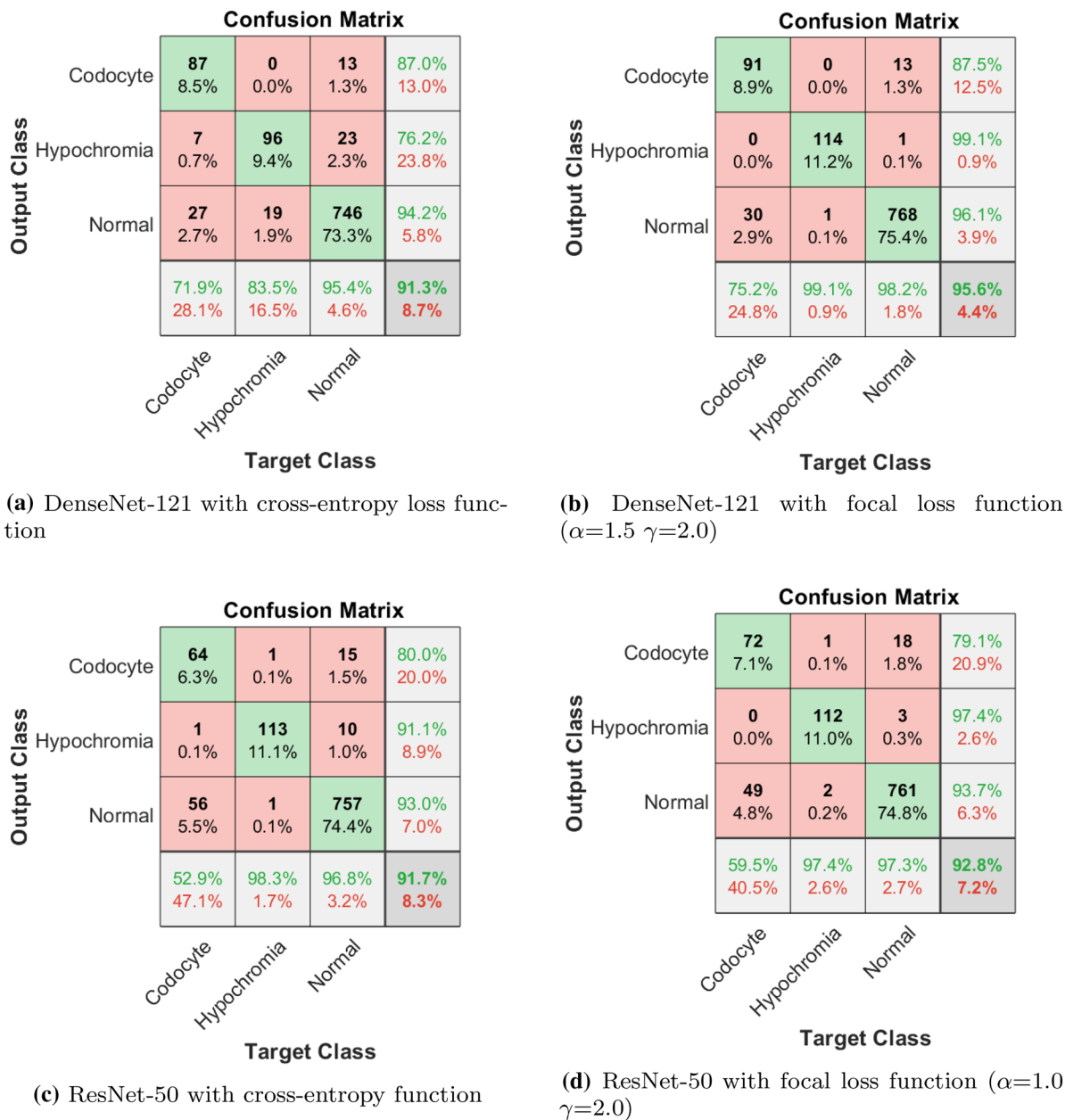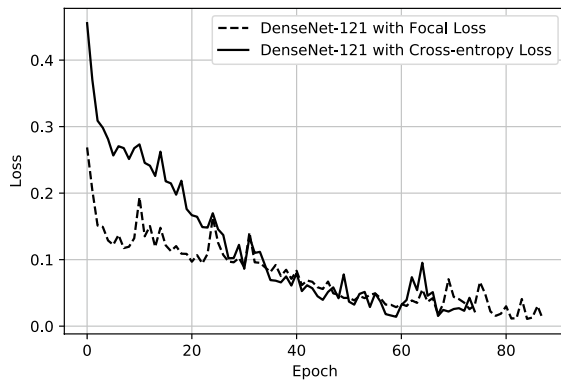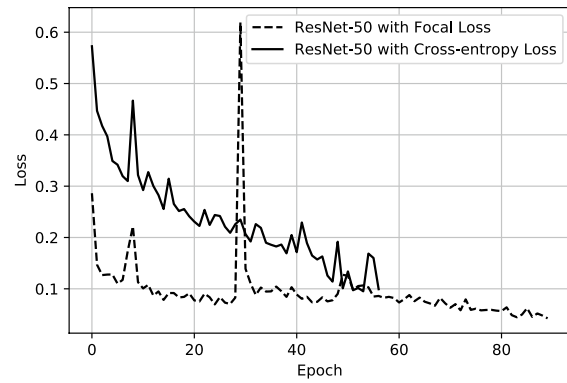(d) ResNet-50 with focal loss function ($\alpha=1.0$ $\gamma=2.0$)

Fig. 11 Comparison of the confusion matrix of both models with either cross-entropy loss function or focal loss function

clearly improved their performances compared to those of both models with cross-entropy loss function. Furthermore, the deep learning models with either kind of loss function outperformed the models with cross-entropy loss function in combination with the under-sampling technique, as expected because of the smaller number training samples. It can be noticed that the use of cross-entropy loss function in combination with the over-sampling technique was able to improve the overall performances of the models to be in line with the performances of both models with focal loss function. DenseNet-121 achieved a 0.92 F_1-score with focal loss function and 0.91 F_1-score with cross-entropy loss function in combination with the over-sampling technique. Hence, employing the over-sampling technique (image augmentation) was truly able to improve the performance of models with cross-entropy functions. This is because over-sampling increased the number of samples in the training phase and successful deep models depended on a large number of samples (Pasupa and Sunhem 2016). It should be noted, though, that performing the over-sampling technique led to a higher computational cost. Overall, incorporating a focal loss function into the models clearly improved their performance.

(a) Comparison of training losses produced by DenseNet-121 with either the focal loss function or the cross-entropy loss function

(b) Comparison of training losses produced by ResNet-50 with either the focal loss function or the cross-entropy loss function

Fig. 12 Comparison of training losses produced by both models with either the focal loss function or the cross-entropy loss function

Next, we discuss the AUROC and confusion matrix of the best model. Fig. 10 shows that both models with focal loss function achieved higher AUROC (near 1) for all classes than those achieved by models with cross-entropy loss function. This indicates that models with focal loss function were able to separate out each class very well in a class imbalance scenario.

In addition, we analysed the confusion matrix of models with either kind of loss function. The results are shown in Fig. 11. It should be noted that two minority classes were Codocyte that made up only 10.17% of the whole samples and Hypochromia that made up only 10.50% of the whole samples. ResNet-50 model with cross-entropy loss function achieved 52.9% accuracy in classifying Codocyte and 98.3% in classifying Hypochromia. The model did not perform well on Codocyte because it misclassified Codocyte to be Normal class for 5.5% of the total number of samples. This implies that the model was biased towards the majority class. However, when focal loss function was applied, the model did better in classifying Codocyte class and Normal class. This improved the model accuracy by 6.6%. Similarly, DenseNet-121 with cross-entropy loss function has an issue with classifying between the two minority classes and the

majority class. DenseNet-121 achieved only 71.9% accuracy in classifying Codocyte and 83.5% accuracy in classifying Hypochromia. This is the common effect of the class imbalance problem. Thus, we incorporated the focal loss function into our model. Then, it was able to differentiate both minority classes more evidently and with high improvement. The accuracy in classifying Codocyte increased by 3.3% and in classifying Hypochromia increased by 15.6%.

Furthermore, we compared the training losses of both models with focal loss and cross-entropy loss functions in Fig. 12. Focal loss function is a generalised version of cross-entropy loss function; therefore, they can be plotted along the same co-ordinate axis. It can be clearly seen that the focal loss function enabled the training loss to converge to zero faster than the cross-entropy loss function could.

Furthermore, we examined the performances of the model with either the focal loss function or the cross-entropy loss function in dealing with class imbalance at different levels, shown in Table 2. The class distribution ratio of the original dataset was 10:10:80. To simulate a higher level of imbalance, we randomly removed samples in the minority classes and fixed the number of samples in the majority class. Conversely, we randomly removed samples in the majority class

Table 2 Class distributions where $|C_i|$ is the number of samples in $C_i : i \in \{0, 1, 2\}$ (C_0 is Codocyte; C_1 is Hypochromia; and C_2 is Normal case)

Class Distributions			ρ	$|C_0|$	$|C_1|$	$|C_2|$	$\sum_{i=0}^{3} |C_i|$
C_0	C_1	C_2					
25.0	25.0	50.0	2.03	345	356	701	1402
20.0	20.0	60.0	3.05	345	356	1052	1753
15.0	15.0	70.0	4.74	345	356	1636	2337
10.0	10.0	80.0	7.80	345	356	2691	3392
7.5	7.5	85.0	11.35	237	237	2691	3165
5.0	5.0	90.0	17.94	150	150	2691	2991
2.5	2.5	95.0	37.90	71	71	2691	2833

Table 3 Comparison of the performances of DenseNet-121 and ResNet-50 with different ρ with either focal loss function or cross-entropy loss function. Both models were run with their optimal hyperparameters

ρ	Model architecture											
	DenseNet-121						ResNet-50					
	Focal Loss			Cross-entropy Loss			Focal Loss			Cross-entropy Loss		
	Accuracy	F_1-score	AUROC	Accuracy	F_1-score	AUROC	Accuracy	F_1-score	AUROC	Accuracy	F_1-score	AUROC
2.03	**89.79**	**0.90**	**0.97**	86.22	0.84	**0.97**	**88.36**	**0.88**	**0.97**	88.36	0.87	0.97
3.05	**91.63**	**0.90**	**0.98**	89.92	0.88	0.97	**89.16**	**0.88**	**0.97**	83.27	0.81	0.94
4.74	**87.61**	**0.84**	**0.95**	87.32	0.83	**0.95**	**90.60**	**0.86**	**0.97**	90.03	0.84	0.95
7.80	**95.60**	**0.92**	**0.99**	91.30	0.84	0.95	**92.8**	**0.87**	**0.97**	91.7	0.85	0.96
11.35	**96.32**	**0.91**	**0.99**	92.00	0.79	0.97	92.11	**0.81**	**0.95**	**93.37**	0.80	0.94
17.94	**96.10**	**0.85**	**0.98**	93.99	0.78	0.97	**94.32**	**0.79**	**0.94**	92.43	0.78	**0.94**
37.90	94.00	**0.59**	0.92	**96.12**	0.57	**0.94**	94.82	**0.56**	**0.92**	94.82	0.51	0.88

The best performance for each model is expressed as bold text

Fig. 13 Average Relative Improvement/Worsening of Accuracy, F_1-score and AUROC of CNN models with focal loss function against CNN models with cross-entropy loss function
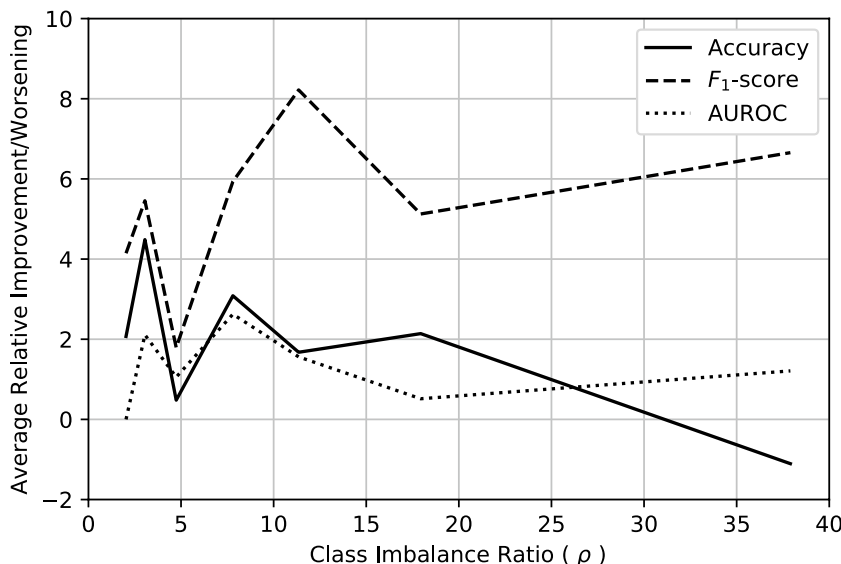
while fixing the number of samples in the minority classes to lower the level of imbalance. We calculated the imbalance ratio (ρ) (Buda et al. 2018) based on the distributions we obtained in Table 2 by

$$\rho = \frac{\max_i(|C_i|)}{\min_i(|C_i|)}, \quad (5)$$

where C_i is the set of samples in class i, $\max_i(|C_i|)$ and $\min_i(|C_i|)$ are the maximum and the minimum number of samples in all classes. Then, we randomly split 70% of each manipulated dataset to be a training set and the remaining 30% to be a test set. The model was then trained with the same settings described in Sect. 4.2.

Employing the focal loss function on DenseNet-121 and ResNet-50 yielded better performances than using cross-entropy loss function for all ρ, resulting in higher F_1-scores as shown in Table 3. Nevertheless, there were some cases that the models achieved a higher F_1-score but a lower accuracy and AUROC, e.g., DenseNet-121 with $\rho = 37.90$ and ResNet-50 with $\rho = 11.35$. Owing to these cases, it can be interpreted that the focal loss function made the deep CNN models learn to classify each class equally without biased towards the majority class. In contrast, the cross-entropy loss function provided a higher accuracy and AUROC score but lower F_1-score, resulting in a biased model towards the majority class.

We calculated and plotted the relative improvement and worsening in classification performance of DenseNet-121 and ResNet-50 achieved by incorporation of either focal loss function or cross-entropy loss function by averaging the relative improvement across both models in Fig. 13. The models

with the focal loss function showed a higher improvement in F_1-score than in accuracy and AUROC. In addition, the relative improvement of F_1-score tended to increase as ρ got higher. In contrast, the relative improvement in accuracy of the models with focal loss function tended to decrease because the biased model with cross-entropy loss function gained a higher accuracy but a lower F_1-score as ρ got higher, i.e., the biased model failed to correctly classify the minority class.

6 Conclusion

Class imbalance is commonly encountered in real-world data, particularly in medical image data. It makes a classification model easily biased towards the majority class. In addition, an inaccurate prediction may lead to false disease diagnosis. Hence, a way to deal with imbalanced data properly is very important. In this work, we proposed a method to solve class imbalance issue by incorporating a focal loss function into deep CNNs. Furthermore, we proposed a well-designed framework for class imbalance learning. We demonstrated that CNN models using a focal loss function achieved a higher F_1-score than the CNNs using a cross-entropy loss function. In addition, DenseNet-121 performed better than ResNet-50 and used fewer model parameters when using a focal loss function.

References

Abulnaga SM, Rubin J (2018) Ischemic stroke lesion segmentation in CT perfusion scans using pyramid pooling and focal loss. In: Proceedings of the International MICCAI Brainlesion Workshop (BrainLes 2018), Granada, Spain, Springer, pp 352–363. https://doi.org/10.1007/978-3-030-11723-8_36

Baloch BK, Kumar S, Haresh S, Rehman A, Syed T (2019) Focused anchors loss: cost-sensitive learning of discriminative features for imbalanced classification. In: Proceedings of the 11th Asian Conference on Machine Learning (ACML 2019), Nagoya, Japan

Buda M, Maki A, Mazurowski MA (2018) A systematic study of the class imbalance problem in convolutional neural networks. Neural Netw 106:249–259. https://doi.org/10.1016/j.neunet.2018.07.011

Castro CL, Braga AP (2013) Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data. IEEE Trans Neural Netw Learn Syst 24(6):888–899. https://doi.org/10.1109/TNNLS.2013.2246188

Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357. https://doi.org/10.1613/jair.953

Chy TS, Rahaman MA (2019) A comparative analysis by knn, svm & elm classification to detect sickle cell anemia. In: Proceedings of 2019 international conference on robotics, electrical and signal processing techniques (ICREST 2019), Dhaka, Bangladesh, pp 455–459. https://doi.org/10.1109/ICREST.2019.8644410

Datta S, Das S (2015) Near-bayesian support vector machines for imbalanced data classification with equal or unequal misclassification costs. Neural Netw 70:39–52. https://doi.org/10.1016/j.neunet.2015.06.005

Doi K, Iwasaki A (2018) The effect of focal loss in semantic segmentation of high resolution aerial image. In: Proceedings of the 2018 IEEE international geoscience and remote sensing symposium (IGARSS 2018), Valencia, Spain, pp 6919–6922. https://doi.org/10.1109/IGARSS.2018.8519409

Dong C, Loy CC, He K, Tang X (2015) Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell 38(2):295–307. https://doi.org/10.1109/TPAMI.2015.2439281

Drummond C, Holte RC et al (2003) C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In: Proceedings of Workshop on Learning from Imbalanced Datasets II, vol 11. Washington, DC, USA, pp 1–8

Durant TJ, Olson EM, Schulz WL, Torres R (2017) Very deep convolutional neural networks for morphologic classification of erythrocytes. Clin Chem 63(12):1847–1855. https://doi.org/10.1373/clinchem.2017.276345

Ford J (2013) Red blood cell morphology. Int J Lab Hematol 35(3):351–357. https://doi.org/10.1111/ijlh.12082

Garcı S, Triguero I, Carmona CJ, Herrera F et al (2012) Evolutionary-based selection of generalized instances for imbalanced classification. Knowl Based Syst 25(1):3–12. https://doi.org/10.1016/j.knosys.2011.01.012

García S, Herrera F (2009) Evolutionary undersampling for classification with imbalanced datasets: proposals and taxonomy. Evol Comput 17(3):275–306. https://doi.org/10.1162/evco.2009.17.3.275

Garcia V, Sanchez JS, Mollineda RA, Alejo R, Sotoca JM (2007) The class imbalance problem in pattern classification and learning. In: Proceedings of II Congreso Espanol de Informatica, pp 283–291

Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the 13th international conference on artificial intelligence and statistics (AISTAT 2010), Sardinia, Italy, pp 249–256

Habibzadeh M, Krzyzak A, Fevens T (2011) Application of pattern recognition techniques for the analysis of thin blood smear images. J Med Inf Technol 18

Haixiang G, Yijing L, Shang J, Mingyun G, Yuanyue H, Bing G (2017) Learning from class-imbalanced data: review of methods and applications. Expert Syst Appl 73:220–239. https://doi.org/10.1016/j.eswa.2016.12.035

Han H, Wang WY, Mao BH (2005) Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In: Proceedings of international conference on intelligent computing (ICIC 2005), Hefei, China, Springer, pp 878–887. https://doi.org/10.1007/11538059_91

He H, Bai Y, Garcia EA, Li S (2008) ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In: Proceedings of the IEEE international joint conference on neural networks (IJCNN 2008), Hong Kong, China, pp 1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969

He K, Sun J (2015) Convolutional neural networks at constrained time cost. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2015), Boston, MA, USA, pp 5353–5360. https://doi.org/10.1109/CVPR.2015.7299173

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2016), Las Vegas, NV, USA, pp 770–778. https://doi.org/10.1109/CVPR.2016.90

Hospedales TM, Gong S, Xiang T (2011) Finding rare classes: active learning with generative and discriminative models. IEEE Trans Knowl Data Eng 25(2):374–386. https://doi.org/10.1109/TKDE.2011.231

Huang C, Li Y, Change Loy C, Tang X (2016) Learning deep representation for imbalanced classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, pp 5375–5384. https://doi.org/10.1109/CVPR.2016.580

Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2017), Honolulu, HI, USA, pp 4700–4708. https://doi.org/10.1109/CVPR.2017.243

Japkowicz N, Stephen S (2002) The class imbalance problem: a systematic study. Intell Data Anal 6(5):429–449. https://doi.org/10.3233/IDA-2002-6504

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. ArXiv Preprint ArXiv:14126980

Krawczyk B (2016) Learning from imbalanced data: open challenges and future directions. Prog Artif Intell 5(4):221–232. https://doi.org/10.1007/s13748-016-0094-0

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436. https://doi.org/10.1038/nature14539

Lee H, Chen YPP (2014) Cell morphology based classification for red cells in blood smear images. Pattern Recogn Lett 49:155–161. https://doi.org/10.1016/j.patrec.2014.06.010

Lim P, Goh CK, Tan KC (2016) Evolutionary cluster-based synthetic oversampling ensemble (eco-ensemble) for imbalance learning. IEEE Trans Cybern 47(9):2850–2861. https://doi.org/10.1109/TCYB.2016.2579658

Lin TY, Goyal P, Girshick R, He K, Dollár P (2020) Focal loss for dense object detection. IEEE Trans Pattern Anal Mach Intell 42(2):318–327. https://doi.org/10.1109/TPAMI.2018.2858826

Ma Y, Sun J, Zhou Q, Cheng K, Chen X, Zhao Y (2018) CHS-NET: A cascaded neural network with semi-focal loss for mitosis detection. In: Proceedings of 10th Asian conference on machine learning (ACML 2018), Beijing, China. pp 161–175

Mazurowski MA, Habas PA, Zurada JM, Lo JY, Baker JA, Tourassi GD (2008) Training neural network classifiers for medical decision making: the effects of imbalanced datasets on classification performance. Neural Netw 21(2–3):427–436. https://doi.org/10.1016/j.neunet.2007.12.031

Pasupa K, Sunhem W (2016) A comparison between shallow and deep architecture classifiers on small dataset. In: Proceeding of the 8th International Conference on Information Technology and Electrical Engineering (ICITEE 2016), Yogyakarta, Indonesia, pp 390–395. https://doi.org/10.1109/ICITEED.2016.7863293

Qin F, Gao N, Peng Y, Wu Z, Shen S, Grudtsin A (2018) Fine-grained leukocyte classification with deep residual learning for microscopic images. Comput Methods Programs Biomed 162:243–252. https://doi.org/10.1016/j.cmpb.2018.05.024

Rahman MM, Davis D (2013) Addressing the class imbalance problem in medical datasets. Int J Mach Learn Comput 3(2):224. https://doi.org/10.7763/IJMLC.2013.V3.307

Razzak MI, Naz S (2017) Microscopic blood smear segmentation and classification using deep contour aware CNN and extreme machine learning. In: Proceedings of 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW 2017), Honolulu, HI, USA, pp 801–807. https://doi.org/10.1109/CVPRW.2017.111

Razzak MI, Naz S, Zaib A (2018) Deep learning for medical image processing: overview, challenges and the future. Classif. BioApps 26:323–350. https://doi.org/10.1007/978-3-319-65981-7_12

Rehman A, Abbas N, Saba T, Rahman SIu, Mehmood Z, Kolivand H (2018) Classification of acute lymphoblastic leukemia using deep learning. Microsc Res Tech 81(11):1310–1317. https://doi.org/10.1002/jemt.23139

Ross NE, Pritchard CJ, Rubin DM, Duse AG (2006) Automated image processing method for the diagnosis and classification of malaria on thin blood smears. Med Biol Eng Comput 44(5):427–436. https://doi.org/10.1007/s11517-006-0044-2

Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M et al (2015) Imagenet large scale visual recognition challenge. Int J Comput Vision 115(3):211–252. https://doi.org/10.1007/s11263-015-0816-y

Srivastava RK, Greff K, Schmidhuber J (2015) Highway networks. ArXiv Preprint ArXiv:150500387

Sudre CH, Li W, Vercauteren T, Ourselin S, Cardoso MJ (2017) Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Proceedings of the 3rd international workshop, deep learning in medical image analysis (DLMIA 2017) and 7th international workshop in multimodal learning for clinical decision support (CDS 2017), QC, Canada, pp 240–248. https://doi.org/10.1007/978-3-319-67558-9_28

Suresh S, Sundararajan N, Saratchandran P (2008) Risk-sensitive loss functions for sparse multi-category classification problems. Inf Sci 178(12):2621–2638. https://doi.org/10.1016/j.ins.2008.02.009

Taherisadr M, Nasirzonouzi M, Baradaran B, Mehdizade A (2013) New approch to red blood cell classification using morphological image processing. Shiraz E-Med J 14(1):44–53

Tian X, Wu D, Wang R, Cao X (2018) Focal text: an accurate text detection with focal loss. In: Proceedings of the 25th IEEE international conference on image processing (ICIP 2018), Athens, Greece, pp 2984–2988. https://doi.org/10.1109/ICIP.2018.8451241

Tiwari P, Qian J, Li Q, Wang B, Gupta D, Khanna A, Rodrigues JJ, de Albuquerque VHC (2018) Detection of subtype blood cells using deep learning. Cogn Syst Res 52:1036–1044. https://doi.org/10.1016/j.cogsys.2018.08.022

Tomari R, Zakaria WNW, Jamil MMA, Nor FM, Fuad NFN (2014) Computer aided system for red blood cell classification in blood smear image. Proc Comput Sci 42:206–213. https://doi.org/10.1016/j.procs.2014.11.053

Wang S, Liu W, Wu J, Cao L, Meng Q, Kennedy PJ (2016) Training deep neural networks on imbalanced data sets. In: Proceedings of 2016 international joint conference on neural networks (IJCNN 2016), Vancouver, BC, Canada, pp 4368–4374. https://doi.org/10.1109/IJCNN.2016.7727770

Weiss GM (2004) Mining with rarity: a unifying framework. ACM SIGKDD Explor Newsl 6(1):7–19. https://doi.org/10.1145/1007730.1007734

Xu M, Papageorgiou DP, Abidi SZ, Dao M, Zhao H, Karniadakis GE (2017) A deep convolutional neural network for classification of red blood cells in sickle cell anemia. PLoS Comput Biol 13(10):e1005746. https://doi.org/10.1371/journal.pcbi.1005746

Yue S (2017) Imbalanced malware images classification: a CNN based approach. ArXiv Preprint ArXiv:170808042

Zhang C, Tan KC, Li H, Hong GS (2018) A cost-sensitive deep belief network for imbalanced classification. IEEE Trans Neural Netw Learn Syst 30(1):109–122. https://doi.org/10.1109/TNNLS.2018.2832648

Zhang X, Fang Z, Wen Y, Li Z, Qiao Y (2017) Range loss for deep face recognition with long-tailed training data. In: Proceedings of the IEEE international conference on computer vision (ICCV 2017), Venice, Italy, pp 5409–5418. https://doi.org/10.1109/ICCV.2017.578

Zhao J, Zhang M, Zhou Z, Chu J, Cao F (2017) Automatic detection and classification of leukocytes using convolutional neural

networks. Med Biol Eng Comput 55(8):1287–1301. https://doi.org/10.1007/s11517-016-1590-x

Zhou ZH, Liu XY (2006) Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Trans Knowl Data Eng 18(1):63–77. https://doi.org/10.1109/TKDE.2006.17

Zong W, Huang GB, Chen Y (2013) Weighted extreme learning machine for imbalance learning. Neurocomputing 101:229–242. https://doi.org/10.1016/j.neucom.2012.08.010