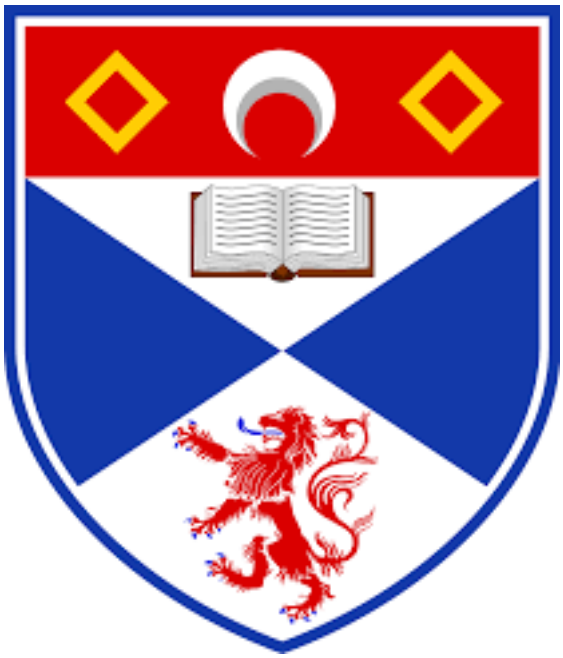
Deep Learning Techniques for Blood Cancer Detection by Cell Classification in Bone Marrow Cellular Images

Lewis Carmichael



Supervised by Dr David Harris-Birtill

BSc (Hons) Computer Science Dissertation Project

University of St Andrews

April 2025

Contents

Abstract

In the UK Blood Cancer is the third biggest cancer killer with around 16,000 people dying each year [1]. Further, for a specialist type of blood cancer called leukaemia over 10,000 new cases were reported each year between 2017-2019 [2]. Computer-aided techniques such as utilising machine and deep learning models for the purposes of cancer detection are needed to reduce work load on radiologists and aid to give patients critical treatment as quickly as possible. This project investigates the use of Convolutional Neural Networks and associated Deep Learning Techniques on the multi-class classification of single-cell images within bone marrow for detecting blood cancer. The BMCMHF dataset [3] comprising over 170,000 cell samples from bone marrow is used to create a deep learning pipeline informed by a Literature Review of related work. In this investigation a previous pipeline is replicated [4] and built upon that utilises a rejection option in the models decision making process. This is an important feature to include when building a deep learning pipeline for medical use, as the decisions made by a model can impact important diagnosis decisions. An extended pipeline is created to utilise the rejected samples in hard negative mining to increase accuracy and reduce rejection rates. From evalaution on the validation set it was found that the replicated pipeline developed, the baseline model, achieved 94.1% accuracy, compared to the [4] pipeline that achieved 99.44%. An updated baseline model to utilise altered hyperparameters and data augmentation achieved 94.7%. Using hard negative mining resulted in an increase to 95.6% accuracy for the baseline model, and 95.8% accuracy for the updated baseline model. The most notable finding was the use of hard negative mining resulted in a rejection rate of 0% whereas the previous implementations resulted in rejection rates of 24% for the baseline model, 27% for the updated baseline model and 31.44% for the pipeline shown in [4]. A decrease in rejection rate results in less samples that would need to be classified, reducing the work load on professionals in practice.

Declaration

I declare that the material submitted for assessment is my own work except where credit is explicitly given to others by citation or acknowledgment. This work was performed during the current academic year except where otherwise stated. The main text of this project report is 15,670 words long, including project specification and plan. In submitting this project report to the University of St Andrews, I give permission for it to be made available for use in accordance with the regulations of the Unversity Library. I also give permission for the title and abstract to be published and for copies of the report to be made and supplied at cost to any bona fide library or research worker, and to be made available on the World Wide Web. I retain copyright in this work.

Acknowledgment

I would like to thank my supervisor David Harris-Birtill for useful advice and support throughout this project which has been invaluable. At the end of my four years of study at St Andrews I want to thank all the incredible staff in the School of Computer Science. I also want to thank all my family and friends for their support in completing this final year of study.

Ethics

This project aims to inbestigate the use of different techniques in machine learning to classify cell types from single-cell images. The project will use secondary data from the The Cancer Image Archive: BMCMHF [3]. No data will be collected, and no surveys or interviews will be required.

Due to the nature of the images being cellular and anonymised, full ethical approval was not required for this project. Further, the data is stored on the encrypted Computer Science School servers and all data processing will be on a School GPU.

The Preliminary Ethics Approval form can be seen in the Appendix.

CHAPTER 1

Introduction

1.1 Background

1.1.1 Artificial Intelligence

Artificial intelligence (AI) refers to technologies that enable machines and computers to simulate human capabilities, including learning, problem-solving, comprehension, and decision-making. Since its inception in the 1950s, AI has continuously evolved, enhancing its capacity to address complex tasks [5]. The emergence of machine learning in the 1980s marked a significant advancement, introducing systems capable of learning from historical data to generate predictions and informed decisions [5]. Among various machine learning algorithms and models, artificial neural networks, inspired by the human brain's interconnected neurons, have become particularly prominent in analysing complex datasets [5].

Supervised learning, a subfield of machine learning, employs labelled datasets where each input sample is paired with an appropriate output label. The objective of supervised learning is for the model to accurately map inputs to their respective outputs, enabling reliable predictions on previously unseen data [5]. Deep learning, a sophisticated subset of machine learning that emerged in the 2010s, utilises multilayered neural networks known as deep neural networks. These structures closely mimic the human brain's decision-making abilities, enabling models to discern intricate features and patterns in data [5]. Consequently, deep learning has found extensive applications across various domains.

1.1.2 Applications of Deep Learning

Machine learning and deep learning techniques have been successfully applied in diverse areas, including fraud detection, image recognition, natural language processing, predictive analytics, and conversational agents (chatbots). Specifically, deep learning has shown exceptional effectiveness in image recognition, notably influencing medical imaging applications [6]. Medical imaging involves generating images of internal body structures for clinical assessment and medical interventions. Radiology, a specialised field within medical practice, extensively utilises imaging technologies for diagnosing and treating diseases and injuries [7]. Radiologists are medical professionals who analyse these images to identify abnormalities and guide patient treatment.

However, human-led image classification tasks, including those in medical imaging, are susceptible to errors due to human limitations. Classification tasks involve analysing an image and accurately identifying its corresponding category. Such tasks may be binary, involving two classes (e.g., tumour versus non-tumour), or multi-class, involving multiple categories (e.g., various types of cells). Radiologists typically perform cell classification manually as part of diagnosing conditions such as blood cancer. This manual process is both labour-intensive and prone to errors [8].

The capability of deep learning models to surpass human accuracy was first notably demonstrated through the ImageNet Large-Scale Visual Recognition Challenge, where models achieved lower error rates than human participants [8]. A breakthrough in medical imaging occurred when Google's deep learning model, GoogLeNet, achieved an accuracy rate of 89% in cancer detection tasks, surpassing the 70% accuracy rate of human pathologists at the time [9]. Such results underline deep learning's significant potential in medical diagnostics.

This investigation will focus specifically on a supervised multi-class classification task using deep learning techniques. The primary objective is to classify single-cell images obtained from bone marrow samples, for the purpose of accurately identifying blood cancer in practice.

1.1.3 Impact of Blood Cancer

In the United Kingdom, blood cancer is the third leading cause of cancer-related deaths, accounting for approximately 16,000 fatalities annually [1]. Blood cancers are broadly categorised into three groups: leukaemia, lymphoma, and myeloma [10]. Leukaemia primarily affects blood cells produced in the bone marrow, especially white blood cells, impairing their ability to fight infections [11, 12]. Lymphoma targets lymphocytes, essential components of the immune system [11]. Myeloma involves abnormal plasma cells that proliferate excessively, potentially forming bone tumours known as plasmacytomas [13], thereby significantly weakening the immune system [12].

Cancer Research UK reported an annual average of 10,302 new leukaemia cases between 2017 and 2019 [2]. In the United States, a diagnosis of leukaemia, lymphoma, or myeloma occurs approximately every three minutes [14]. Globally, around 1.24 million blood cancer cases are diagnosed annually, accounting for about 6% of all cancer diagnoses [15]. In 2020 alone, there were approximately 474,519 new cases of leukaemia worldwide, resulting in 311,594 deaths [16], highlighting the significant global impact of blood cancers.

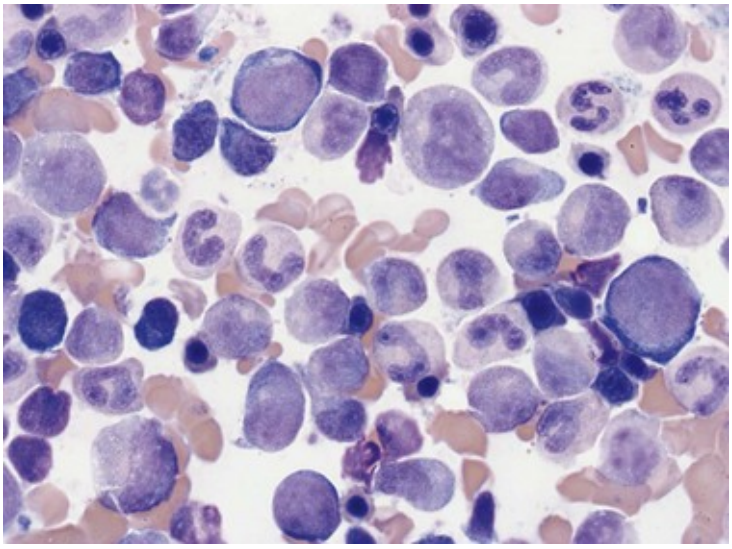


Figure 1.1: *Normal adult haemopoiesis in a bone marrow aspirate stained with May-Grünwald-Giemsa (magnification x400). Taken from: [17].*

1.1.4 Traditional Methods of Blood Cancer Diagnosis

Blood cancer originates from the bone marrow, where blood cells are produced [12]. Cancerous cells typically result from abnormal and uncontrolled proliferation, particularly affecting white blood cells [12]. The standard diagnostic methods for most leukaemias, including acute lymphocytic leukaemia (ALL), acute myeloid leukaemia (AML), and chronic myeloid leukaemia (CML), involve simultaneous procedures of bone marrow biopsy and bone marrow aspiration [18]. For lymphoma, diagnosis typically involves biopsy and subsequent cellular analysis [12]. These procedures gather solid and liquid bone marrow samples, respectively, for detailed microscopic examination.

The analysis of cellular imagery from these samples, an exmaple of which shown in Figure 1.1, is essential for diagnosis, particularly for aggressive cancers like AML, where early detection directly impacts patient outcomes. Hence, deep learning offers a valuable, potentially transformative approach to medical imaging analysis.

1.2 Objectives of Investigation

1.2.1 Primary Objectives

- Literature Review - analyse existing research and discuss the advantages and disadvantages for different approaches to detect blood cancer from cellular images using deep learning models, and find research gaps to be investigated further.
- Deep Learning Pipeline - create a Deep Learning Pipeline, possibly using a CNN architecture to classify cellular images and form the basis for further research with different methods.
- Rigorous Investigation - investigate one of the research gaps discovered from the Literature Review to improve or conduct a new experiment by changing the Deep Learning Pipeline and report on the impact of the results.

1.2.2 Secondary Objectives

- Investigate the impact of using Data Augmentation techniques on the training data and report the results on model performance.

CHAPTER 2

Literature Review

The Literature Review section of this investigation performs an evalaution and description of similar and related work in this field of research, discussing advantages and disadvantages for different approaches to detect blood cancer using deep learning models, identifying any research gaps.

Deep learning techniques have been shown to be effective for image analysis tasks [19] by achieving high accuracy results, and outperforming humans to correctly classify images. This has led to use within many fields including medical imaging. For blood cancer, typical diagnosis processes are time-consuming and error-prone [8] so utilising deep learning in this process will reduce the workload on professionals and ensure improved performance, leading to more accurate diagnosis and identification of diseases for patients [8].

A deep learning architecture, called a Convolutional Neural Network has proven to be a popular and successful choice for general image classification tasks [20–22] as well as in medical imaging tasks [23, 24] by achieving high accuracy results. This investigation will focus on and utilise Convolutional Neural Networks to perform the multi-class classification task using the BMCMHF dataset [3].

2.1 Background: Convolutional Neural Network

Convolutional Neural Networks (CNNs), a deep learning architecture have been shown to provide great performance for general image classification tasks, outperforming other machine learning techniques [20–22]. Further, for image classification within medical imaging, CNNs have been widely adopted in this field [23, 24] due to their excellent feature learning capabilities [23]. Consequently, CNNs provide a robust base-model for use within this research.

2.1.1 Image Classification

A CNN is a type of deep learning algorithm that uses multi-layered neural networks to learn from vast amounts of data [25], and although CNNs can be utilised for other tasks, the primary application has been in image analysis [19].

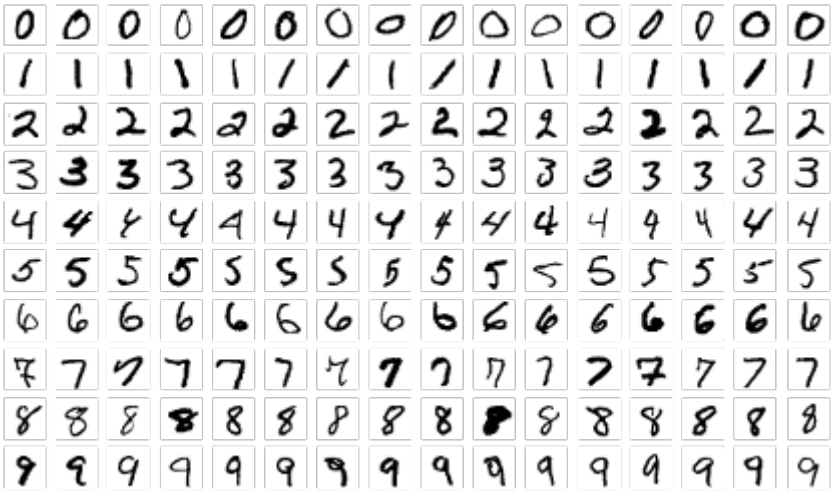


Figure 2.1: *Handwritten digits from the MNIST dataset. Taken from [22].*

The MNIST (Modified National Institute of Standards and Technology) is a dataset composed of images that are handwritten digits (0–9) [22], as shown in Figure 2.1. The aim is to correctly classify the input image into the appropriate category, i.e. the correct number. Utilising a CNN model to complete this task can achieve results exceeding 98% with just a single convolution layer [22]. Moreover, a CNN pipeline achieved 99.6% accuracy using the same MNIST

dataset [26]. These implementations employed pre-processing techniques such as data augmentation and data normalisation to enhance overall accuracy.

The high performance achieved in general classification tasks such as the MNIST [22] task highlighted CNNs ability to be utilised in many domains of work, including medical imaging (see Section 2.2).

2.1.2 CNN Architecture

In the CNN architecture there are multiple layers—the three main types being the Convolutional layer, Pooling layer, and Fully Connected layer [27]. For image classification, a CNN extracts features from the input image to identify patterns established in the dataset [28]. CNNs use filters to capture intricate details and spatial features in the image [28]; these filters are represented by matrices of integers.

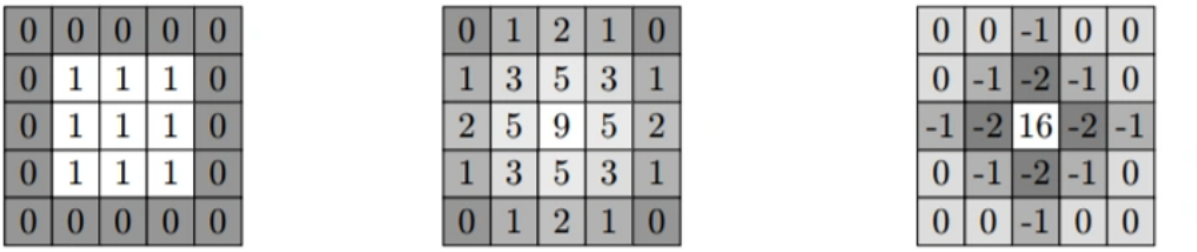


Figure 2.2: *Example of Filters. Taken from [28].*

The Convolution Layer uses filters—also called kernels—that slide across the height and width of the receptive field (the selected portion of the input image), creating a dot product between the filter and the image segment. It is important to note that the spatial size of the filter is smaller than that of the receptive field. The two-dimensional output produced by this operation is known as the activation map, which provides the response of the filter at each spatial position [29].

Subsequent layers reduce the spatial size of the activation maps produced earlier [28], preserving the most important information through downsampling operations—a process referred to as pooling [29]. One or more fully connected layers then flatten the output from the final pooling layer and learn complex relationships between the extracted features, ultimately producing class probabilities or predictions [28].

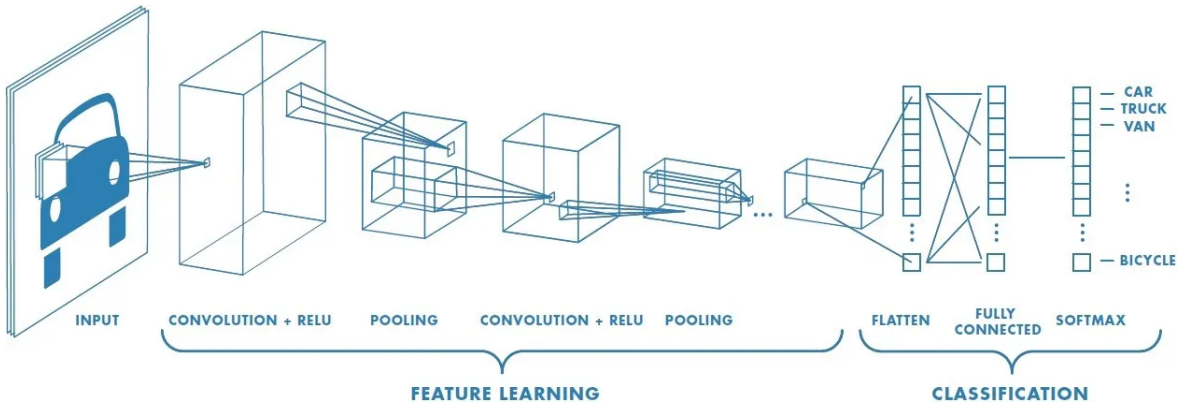


Figure 2.3: *Convolutional Neural Network process for image classification, taken from [30].*

Overall, the CNN works by breaking the input image into small pieces focusing on the features to detect patterns [31]. Earlier layers focus on simple features, such as colours and edges. As the image data progresses through the layers of the CNN, it starts to recognise larger elements or shapes until finally identifying the object [27].

2.2 CNN for Medical Imaging

The use of CNN models in medical imaging had a breakthrough when the GoogLeNet model was used to detect cancer at an accuracy of 89%, while at the time human pathologists achieved an accuracy of 70% [9], demonstrating its potential to be a significant contributor to future medical application. CNNs proficiency in medical image analysis highlights a key application for use within the field medical imaging for tasks such as disease classification, organ region segmentation and localisation and detection of pathological targets [32, 33] which assist clinicians in efficient and accurate image processing.

Many different tasks can be completed using deep learning techniques in medical imaging. This investigation will focus on the classification of single-cells within bone marrow samples. This type of task uses samples of single-cell images attached with labels to performing training and then evaluation on a validation set, a portion of the data set aside for assessing the performance on unseen data.

2.2.1 CNN for Single-cell Bone Marrow Classification

For classifying single-cell bone marrow smear images, in research a widely used technique is to utilise a Deep Learning model with a CNN-based architecture, as shown in multiple research cases below.

In research [34] two CNN-based classifiers for single-cell images of bone marrow leukocytes are created. The first a simple sequential network architecture, and the second using ResNeXt, a deep learning model uses the basic concepts of the ResNet (Residual Network) model that uses several parallel pathways each capturing different features from the input data [35]. The ability to vary depth of ResNeXt network allows for flexibility in optimising the architecture for different tasks [35]. The research [34] utilises the same dataset BMCMHF that will be utilised for our research, as can be seen in Section 1.4.1 [3]. The training-testing split was 80%/20% respectively for the images available for each class, assigned randomly. After which pre-processing techniques such as data augmentation and five-fold cross validation are completed (see Section 2.5). The data augmentation was used to counter class imbalance in the training process (see Section 2.5), the strategy implemented upsampled the training data to 25,000 images per class by performing a set of geometric and stain-colour augmentation transformations. This research utilised the state-of-the-art CNN at the time of publishing for the purpose of morphological classification. The model proposed achieved high precision and recall values for most diagnostically relevant classes [34]. When directly comparing to the feature-based classifier approach mentioned [36], the CNN network clearly outperforms it [34]. This research was the first to utilise the BMCMHF dataset [3] for the use of cell classification, and one of the key findings

was that deep learning-based classification tasks have outperformed other methods that need extraction of handcrafted features [34].

Following on, further research for classifying single-cell bone marrow smears was conducted also using the BMCMHF dataset [3] which introduced a custom-made CNN-based architecture named BoMaCNet [37]. A main objective of this research is to accurately identify and classify some of the most common types of bone marrow cells, thus only the six most common and most important cells were used as the base [37]. A total of 96,000 images were utilised with 16,000 samples being selected from each label. Similarly, the training set was 80%, evaluation 10%, and testing 10% of the available data. Further pre-processing techniques were adopted such as image resizing, geometric augmentation, brightness and contrast augmentations as well as data normalisation [37] (see Section 2.5). The BoMaCNet model design consists of 7 convolution layers and 5 dense layers. A dense layer refers to a layer that is deeply connected with its preceding layer, which means that all the nodes of the layer are connected to every node in the layer before [38]. The model was designed to perform batch normalisation [37] after each convolution, which speeds up the learning procedure for the network [37]. Following the batch normalisation layer, it performs an average-pooling process with a 2x2 filter. It was found that the use of average pooling generated the most successful output as it was able to identify images in the model better than other variations of pooling architectures. Average pooling is a technique where the arithmetic mean of elements in each pooling region is calculated instead of opting for the most activation [39]. The BoMaCNet model overall achieved a training accuracy of 95.71% and a validation accuracy of 93.06%. Furthermore, multiple pre-trained models were compared to BoMaCNet and none of them outperformed the model in any of the testing metrics, showing that compared to some other models available the BoMaCNet model performs the best. Even though the BoMaCNet model performs highly, one limitation to only being limited to classifying six classes is in general practice this model would not be applicable as there are many more than six classes of cells in bone marrow, and use could lead to false classification results being produced in a wider context.

Another interesting implementation for classifying single-cell bone marrow slide images using the BMCMHF dataset [3], shows a Siamese Neural Network model implemented [40] achieving 91% training accuracy and 84% validation accuracy. One of the main objectives in this research was to implement a deep neural network model that eradicated the class imbalance problem. The key feature of the Siamese model is that it works well even when there is a shortage of data available [40]. The Siamese network, often called twin neural network uses identical training parameters for both the given inputs for accomplishing the classification tasks [40]. From the given pair one of the inputs has already been assigned with a particular label, resulting in limited work for the network to extract identical or differentiating features out of the pair. The Siamese model proposed outperformed other CNN-based architectures that were created, CNN+SVM and CNN+XGB that resulted in 32% and 28% accuracy respectively. One of the main advantages discovered for the Siamese Neural Network is that it focuses on the similarity and dissimilarity amongst images between the same and different class labels, respectively. Meaning it does not rely on the feature extraction from the single image itself [40].

Also utilising the BMCMHF dataset [3], research [41] shows improved results in performance achieved compared to the BoMaCNet model [37]. The research [41] investigated the classification of single-cell bone marrow smears using three different models, ResNet50, DenseNet121, and EfficientNet to perform the task. Similarly, pre-processing techniques such as data augmentation and image resizing were implemented. Notable findings include DenseNet121 achieving the most accurate result of 98% and ResNet50 achieving 97% accuracy. DenseNet121 achieved 99% training accuracy, 98% test accuracy, 98% precision, 0.98 recall and an F1 score of 0.98.

The DenseNet121 model is utilised to overcome a challenge faced by a limitation of the conventional CNN design, as the number of total layers increase the vanishing gradient issue arise. Namely the “Densely Connected Convolutional Network” uses an approach that alters the configurations of the connectivity pattern between layers to overcome this problem. Within a DenseNet architecture all layers are connected to every other layer, creating a large network of connections as is shown in Figure 2.4.

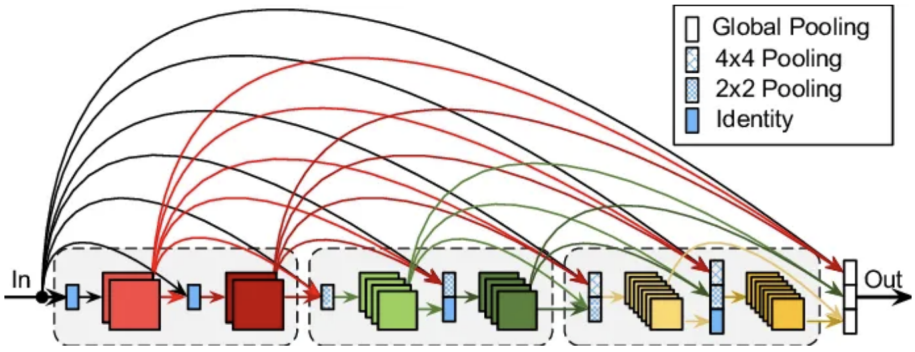


Figure 2.4: *DenseNet-121 architecture. Taken from [41].*

2.2.2 Single-cell Bone Marrow Classification With Reject Option

Due to the nature of image classification in a medical capacity for diagnosis being a highly sensitive decision, it is vital that the results are highly accurate and don't provide incorrect classifications as this could have a knock-on impact to the diagnosis and any treatment or procedures that could be undertaken. Some models created in research implement a reject option [4] [42]. The rejected option is invoked if there is not enough confidence in a classification result, in practice if this was to occur the classification would be left to be handled manually by a professional. A rejected sample is not used within a models evalaution, only accepted samples are utilised. Even though the reject option relies upon human input for rejected cases which could decrease the efficiency of the overall process and increasing the overall time taken, it produces very high accuracy results as shown for model [42] achieving more than 98% accuracy for unrejected cell images. Furthermore, model [4] reaches an accuracy value of over 99% for accepted samples. These high levels of accuracy increase the confidence in the results produced.

2.2.3 State-of-the-art: Single-cell Bone Marrow Classification With Reject Option

For the task of classifying single-cell bone marrow smear images using a rejection option the DenseNet161 model pretrained on ImageNet [43] is the best performing, state-of-the-art model at this time for all research conducted [4] achieving 99.44% accuracy rejecting 31.44% sam-

ples. The investigation outlined in [4] performs a similar task, however utilising a different dataset with around 16,000 images across 11 classes [4].

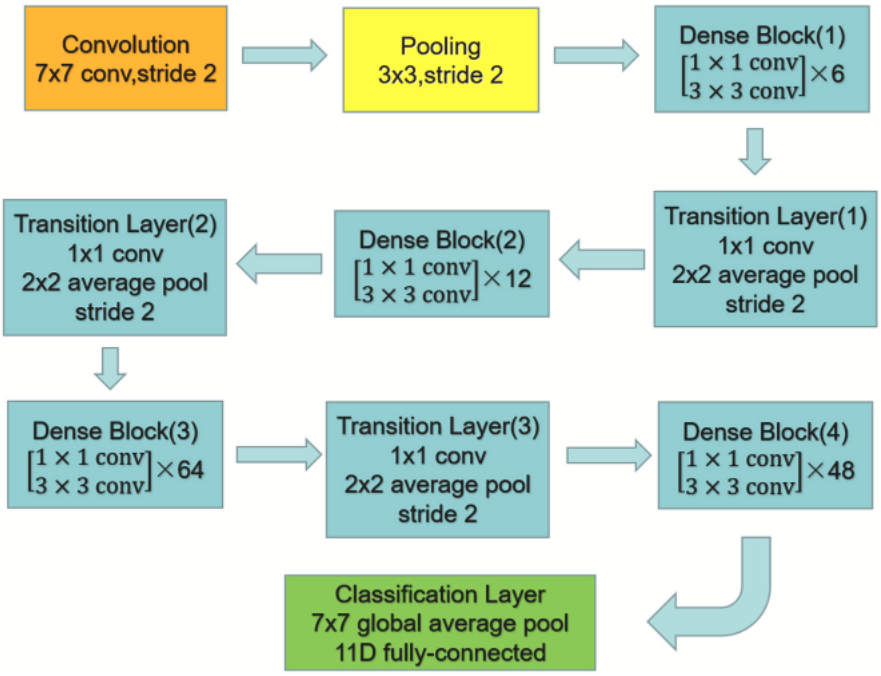


Figure 2.5: *DenseNet-161 architecture. Taken from [4].*

The DenseNet161 architecture shown in Figure 2.5 builds upon the DenseNet121 architecure explained previously in Figure 2.4. The difference between the two model implementations is the depth and complexity of the model, DenseNet121 includes 121 layers and DenseNet161 161 layers. The additional layers within the DenseNet161 moodel result in more parameters, higher computational cost, and results in higher accuracy possible due to more complex features that can be learned [4]. The succesful results achieved within the research in [4] shows an appropriate model to provide a baseline for further investigation.

A research gap identified within implementing the rejection option to the models architecture is not utilising the rejected images after the accept/reject process or during training. After training has converged rejected samples are ignored, only utilising accepted samples in evaluation. At the time of conducting this research no implementation was found that utilised the rejected cases in any part of the training process.

2.2.4 Advantages of CNNs in Medical Imaging

Convolutional Neural Networks (CNNs) have been instrumental in advancing medical image analysis due to their effectiveness in feature extraction and image classification. Their integration into medical diagnostics has demonstrated considerable potential, though significant limitations remain that must be addressed for widespread clinical adoption.

One of the primary strengths of CNNs lies in their ability to autonomously learn hierarchical features from imaging data, making them well-suited for complex diagnostic tasks in medical imaging. Unlike traditional machine learning approaches that often require extensive manual feature engineering, CNNs automatically discern relevant patterns within images, from simple edge detection in initial layers to complex structures in deeper layers [44]. This capability is particularly beneficial in medical imaging, where discerning fine-grained details is critical for diagnostic accuracy [45].

Furthermore, CNNs have demonstrated notable diagnostic accuracy across various medical imaging modalities. For instance, in histopathology and radiology, CNN models have surpassed human-level accuracy in specific diagnostic tasks, such as cancer detection, where models like GoogLeNet achieved an accuracy of 89%, compared to human pathologists' 70% [46]. Such outcomes underscore CNNs' potential as a reliable diagnostic aid, offering consistent and objective analysis that can enhance the reliability of radiological assessments [47].

Additionally, the flexibility of CNN architectures allows them to be tailored to diverse imaging modalities, including MRI, CT, and X-ray, thereby broadening their applicability across medical domains. This adaptability has enabled CNNs to address various clinical tasks, from disease classification and organ segmentation to anomaly detection and localisation [48]. By reducing diagnostic variability and improving speed, CNNs have shown significant promise in

supporting clinicians in high-volume settings, thereby enhancing overall diagnostic efficiency.

2.2.5 Limitations of CNNs in Medical Imaging

Despite these advantages, CNNs are highly data-dependent, requiring extensive labelled datasets for training to achieve optimal performance. Medical imaging datasets often suffer from limited size due to the labor-intensive nature of annotation, privacy constraints, and the rarity of certain disease classes. This data scarcity can lead to overfitting, where the model performs well on training data but generalises poorly to new cases [49]. The issue is exacerbated in medical imaging, where data imbalance is common, as rarer disease classes may be underrepresented, further impairing model performance in clinical settings [50].

In addition to data requirements, CNNs are computationally demanding, particularly in deep architectures with millions of parameters. Training these models requires significant processing power, often necessitating high-performance GPUs and extensive memory, which may be inaccessible in smaller healthcare institutions [47]. These computational limitations not only restrict CNN adoption in resource-limited settings but also present environmental and economic concerns due to the energy consumption associated with large-scale training processes [51].

Moreover, CNNs have been criticised for their ”black-box” nature, as their internal decision-making processes are not easily interpretable. In high-stakes fields like healthcare, understanding the rationale behind diagnostic outputs is critical for building clinician trust and ensuring regulatory compliance [52]. The opaqueness of CNN models hinders transparency, posing challenges for troubleshooting, debugging, and model improvement, particularly when unexpected or erroneous predictions arise. Consequently, this lack of interpretability remains a significant barrier to their clinical integration, as clinicians and patients require insights into how decisions are reached [53].

Lastly, CNNs are sensitive to minor perturbations or inconsistencies in input data, which can lead to variability in outputs. Even subtle changes, such as image noise or variations in acquisition conditions, can impact CNN predictions, potentially leading to misdiagnoses in clinical practice [54]. This sensitivity to input variations necessitates standardized imaging protocols and extensive pre-processing, which may not always be feasible in real-world clinical settings where imaging conditions can be less controlled.

2.3 Background: Vision Transformers

Vision Transformers (ViTs) represent a recent advancement in deep learning architectures for image analysis, emerging as a powerful alternative to Convolutional Neural Networks (CNNs), particularly in applications that benefit from a global context. Originally introduced in natural language processing (NLP), Transformers utilise self-attention mechanisms that allow the model to assess the relative importance of various parts of the input sequence. In the context of image processing, ViTs adapt this approach by dividing images into fixed-size patches, treating each patch as a "token" in a sequence. This unique ability to process relationships between distant image regions enables the model to capture long-range dependencies, offering a different approach to image representation compared to the local feature extraction characteristic of CNNs [55].

Transformers have gained traction in image classification due to their capacity to maintain a holistic view of the image. Unlike CNNs, which process data in a hierarchical, localised manner, ViTs analyse the entire image context from the outset. This attribute allows them to excel in tasks where an understanding of spatial relationships at multiple scales is beneficial, such as recognising complex structures in medical imaging [56].

2.3.1 Vision Transformers in Image Classification

Vision Transformer (ViT) [57] have been proven to outperform CNN in some image classification tasks [58], leading to the potential for a more effective approach to cell image classification. In contrast to CNNs, which rely on convolutional layers to progressively extract features, ViTs apply self-attention across all tokens (i.e., image patches), allowing the model to weigh the significance of each region in the context of the entire image. This approach enables ViTs to capture global contextual information more effectively than CNNs, which are inherently focused on localised features [59].

Studies indicate that ViTs can outperform CNNs in scenarios where the spatial relationships among all parts of the image are critical, and data volume is sufficient to support the complexity of the model. For example, in cell image classification, ViTs have shown the potential to extract more meaningful and global representations of the cell structure, leading to more accurate classification results. Nevertheless, challenges remain in achieving a balance between global and local feature extraction, as ViTs, by design, may overlook fine-grained details necessary in high-resolution medical imaging unless specific adjustments are made [60].

2.3.2 Vision Transformers in Bone Marrow Cell Classification

To address the limitations of standard ViTs in medical applications requiring both global and localised feature extraction, a variant CrossFormer (CF) model was proposed [60]. The CF model introduces an architecture that integrates cross-scale attention mechanisms, effectively capturing both fine-grained and larger structural patterns within images. This is particularly beneficial in bone marrow cell classification, where understanding both the morphology of individual cells and their contextual relationships is essential for accurate diagnosis [61].

The CF model enhances ViT performance in bone marrow cell recognition by employing multi-scale attention, allowing the model to prioritise relevant features at varying resolutions. This

multi-scale approach has demonstrated substantial improvements in classification accuracy and is well-suited to tasks that require identifying subtle morphological differences between cell types. For instance, by integrating cross-scale information, the CF model significantly increases classification precision, addressing a critical gap in ViT's application to medical imaging [62].

2.3.3 Advantages of Vision Transformers

Vision Transformers bring several advantages to medical imaging, yet they also present notable challenges. One primary advantage is their ability to capture global dependencies across an image, which is particularly valuable in medical tasks where the spatial relationships among distant features are diagnostically significant. This global perspective allows ViTs to outperform CNNs in certain large-scale classification tasks, enhancing diagnostic precision and enabling more sophisticated pattern recognition in complex medical datasets [56].

2.3.4 Limitations of Vision Transformers

However, ViTs exhibit limitations that are especially pertinent in medical imaging. First, they demand large datasets to perform optimally. In contrast to CNNs, which can achieve reasonable performance on smaller datasets through convolutional feature reuse, ViTs are prone to overfitting when data is limited due to their extensive parameterisation. This dependency poses a challenge in medical imaging, where labelled data is often scarce, expensive to annotate, and may lack diversity, particularly for rare diseases [49].

Additionally, Vision Transformers are computationally intensive, necessitating substantial processing power and memory, which may be impractical in clinical environments with limited computational resources. This increased computational load can hinder real-time analysis, a crucial requirement in certain medical settings. Furthermore, while ViTs excel at capturing global patterns, they may struggle with finer, localised features without substantial modifications, making them less suitable for applications that demand high-resolution detail, such as

cell morphology analysis or tumour margin delineation [60].

In summary, while Vision Transformers represent a promising advancement in medical image classification, especially for applications where a global understanding of image context is critical, their limitations in terms of data requirements, computational demand, and potential neglect of localised features warrant careful consideration. Future research should explore hybrid architectures or modified attention mechanisms that can balance ViT's global perspective with the localised precision needed in medical diagnostics.

2.4 Dataset

2.4.1 Bone Marrow Cytomorphology MLL Helmholtz Fraunhofer

The Bone Marrow Cytomorphology MLL Helmholtz Fraunhofer (BMCMHF) dataset will be the primary source of single-cell images for my tasks. Containing over 170,000 de-identified, expert-annotated cells from the bone marrow smears of 945 patients stained using the May-Grünwald-Giemsa/Pappenheim stain [3]. The diagnosis distributions in the cohort included a variety of hematological diseases reflective of the sample entry of a large laboratory specialised in leukaemia diagnostics. Image acquisition was performed using a brightfield microscope with 40x magnification and oil immersion. This dataset is one of the most extensive in terms of number of patients, diagnoses, and single-cell images included.

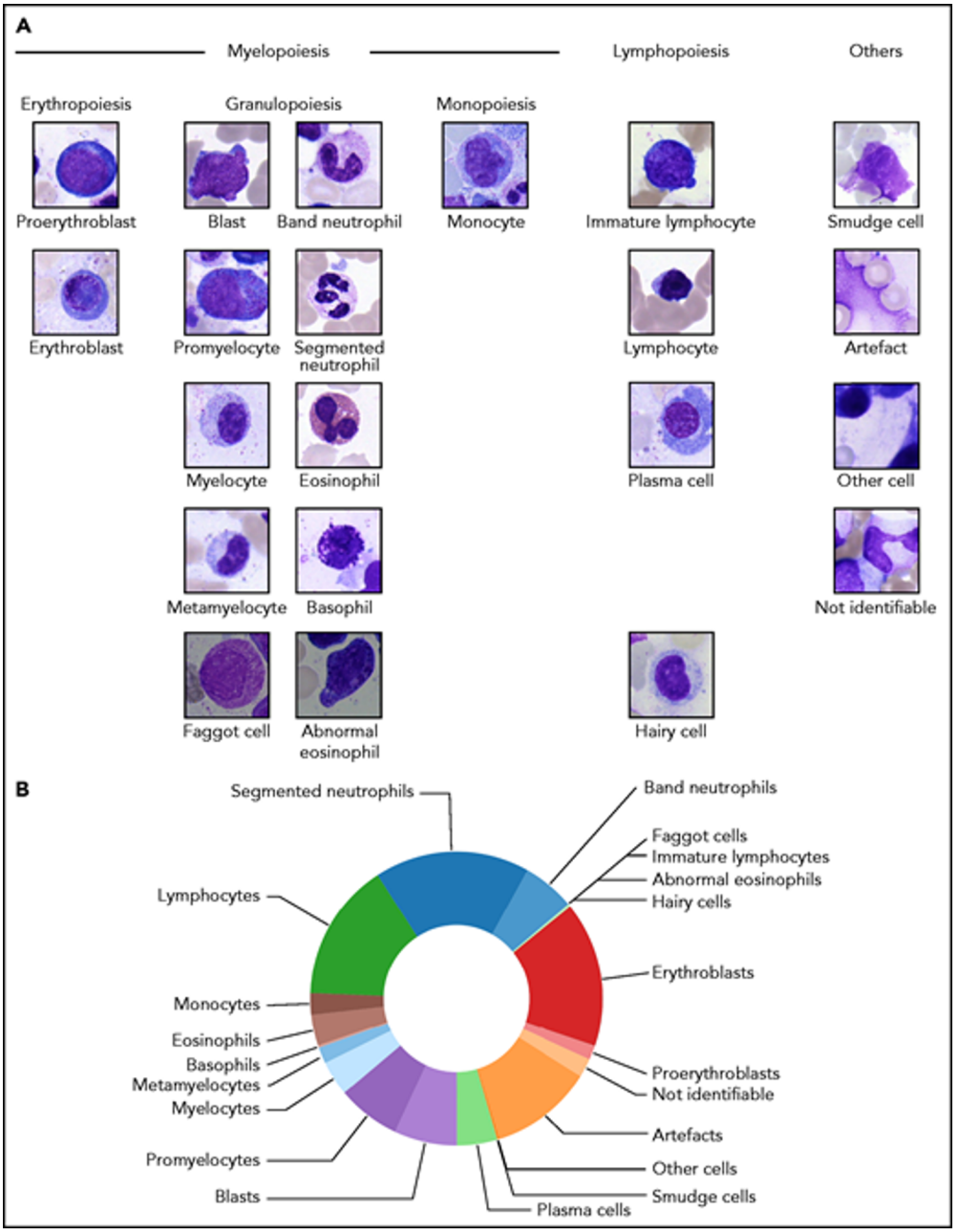


Figure 2.6: *Structure of the 21 morphological classes of BM cells from the dataset. Taken from [3, 34].*

2.4.2 Class Imbalance Problem

It is a common occurrence for there to be an imbalance in the number of images for each class present in the medical image collections, with most classification methods assuming equal occurrences of classes [63]. Shown in Figure 6 in the BMCMHF dataset [3] there exists an imbalance in the number of images available. Furthermore, there are several models that perform well using a balanced dataset but perform worse when using an imbalanced counterpart [64].

Without sufficient data for each class the model will struggle to train appropriately. Specifically related to medical imaging this problem can impact the detection of outliers, and rare health occurrences [63]. Due to a cell image classifier making decisions with potentially highly sensitive outcomes, it is important that this issue is mitigated as best as possible. To eradicate this problem a technique can be adopted called oversampling [65]. This could be data augmentation (see Section 2.5) to increase the total number of available images to use for certain classes.

2.5 Pre-processing Techniques

2.5.1 Data Augmentation

To counteract the class imbalance problem than can occur in medical image collections [63] a pre-processing technique called data augmentation can be used to increase the training sample and lessen the impact of class imbalance. Different augmentation techniques can be adopted to increase the data available for particular classes, this could be through geometric augmentation where the image is flipped, rotating the image left and right certain amounts as performed in research [37]. Moreover, another augmentation technique used is to alter an images contrast and brightness, known as colour image processing [66]. Adding to an images noise is known an intensity transformation [66] and this impacts the image at the pixel or patch level and introduces random noise to the image. Further image augmentation includes cropping to select a particular region of interest within an image with the rest being discarded, shearing an image involves shifting one part of the image along a parallel line, distorting its shape, and translation which refers to shifting an image along its spatial dimensions [67].

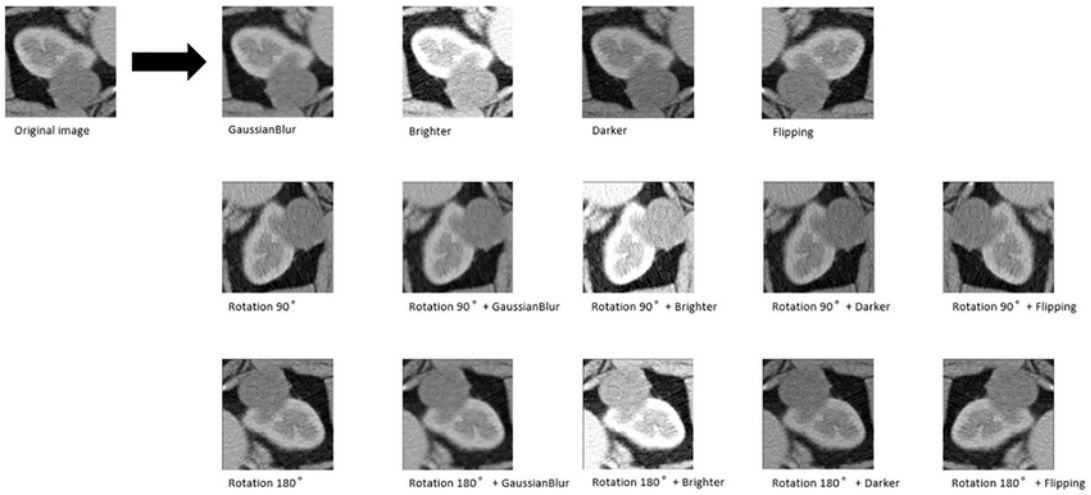


Figure 2.7: *Examples of different data augmentation techniques. Taken from [68].*

Data Augmentation is a technique used to stop the model from overfitting [69] and to generalise the data [67]. A model will overfit if it fits too closely to the training data provided, this can be avoided by training with more data, data augmentation [70] this ensures the model is exposed to a wide range of different versions from the same class. An overfitting model for a certain class of images would likely result in high accuracy for very similar images as what it was trained on, however a slight deviation could result in an incorrect classification. Data Augmentation does present some drawbacks however, by adding to the amount of overall data for training of the model this adds to the computational costs, if data augmentation is poorly executed it will result in too much noise being introduced to the data which could lead to decreased performance on test set [71]. When correctly implemented data augmentation has been shown to improve a model's performance [71]. Additionally, this is important to increase the generalisability of a model [72] to make it more applicable for use within general medical practice. When a model is trained on a single dataset, there is a risk of the model overfitting [49], learning the training data in too much detail. This gives rise to a problem relating to a model's generalisability, an overfitting model could struggle to generalise to a different dataset with different image protocols [72]. For a model to be applicable in medical practice it would need to be able to generalise well to different datasets.

2.5.2 Data Normalisation

A further pre-processing technique that can be used is Data Normalisation to stabilise the model and help with convergence, as a result inconsistencies within the data were removed [37]. The process of Data Normalisation traditionally rescales the original data, so all data lies within the same scale [73]. In the case of image data, the aim is to minimise the bias of features whose contribution is higher in discriminating pattern classes [74]. However, this is not always an effective technique as seen in [74] which shows some methods complicate the normalised data more compared to the un-normalised data. Furthermore, in the same research it is highlighted that not one method of Data Normalisation fits all, as it ultimately depends on the model and dataset being utilised.

2.5.3 Image Resizing

Image resizing is a standard pre-processing technique that ensures images fed into the model have uniform dimensions. Consistent image dimensions allow the model to maintain the expected input size, improving training stability. Techniques such as interpolation and downscaling are commonly used; however, resizing can introduce distortion if the original image aspect ratio is not preserved, potentially impacting classification accuracy in medical contexts where precise features are critical for diagnosis.

2.6 Post-processing Techniques

Post-processing techniques play a critical role in validating and refining machine learning models to ensure they perform robustly across diverse data distributions. These techniques are particularly important for model generalization and evaluation, as they help prevent overfitting to training data while providing a more comprehensive understanding of model performance. By applying post-processing techniques, researchers can enhance model interpretability and improve the reliability of results, which is essential in academic and applied machine learning research.

2.6.1 Five-fold Cross Validation / K-fold Cross Validation

K-fold cross-validation, especially its five-fold variant, is a widely adopted technique in machine learning for enhancing model evaluation reliability. This approach partitions the dataset into five equal sections or "folds." In each iteration, one unique fold is designated as the test set while the remaining four folds serve as the training set, thus generating five separate models. After each model is trained and tested on a different fold, the results are averaged across the five runs, yielding a more robust estimate of the model's generalisation performance [**?**]. By evaluating the model on multiple subsets of the data, five-fold cross-validation reduces the likelihood that a single train-test split will introduce bias or fail to represent the model's true performance, as each data point is used once for testing and four times for training.

Five-fold cross-validation strikes a balance between computational efficiency and statistical reliability. While higher values of KK (e.g., 10-fold) might provide slightly more stable estimates, five-fold cross-validation has been shown to sufficiently mitigate issues of overfitting and underfitting while reducing computational cost [**?**]. This method is especially beneficial when data is limited, as it maximises data usage without necessitating a large test set. Consequently, five-fold cross-validation is widely employed across machine learning research and applications as an essential post-processing technique that supports reliable model selection and performance evaluation [**?**].

2.6.2 Threshold Adjustment

Threshold adjustment is a method used in medical imaging classification models to calibrate decision boundaries based on model confidence. This approach helps in minimising false positives or false negatives by adjusting the threshold at which predictions are considered positive or negative. For instance, in bone marrow cell classification, a higher threshold may reduce false positives, enhancing reliability, but may also result in missed diagnoses if the threshold

is set too conservatively. This technique is particularly valuable in clinical applications where model confidence is as critical as model accuracy.

This technique is a tool that can be utilised to implement a reject option within a deep learning pipeline, similar to the state-of-the-art implementation shown in that uses a softmax function to determine whether to accept or reject samples [4].

CHAPTER 3

Methodology

3.1 Approach

This investigation's initial approach was to replicate the deep learning pipeline shown in the [4], which compared multiple CNN-based architectures that used an accept/reject mechanism during evaluation after the model's training had converged. The research conducted in [4] investigated using CNN-based architectures such as GoogLeNet [9], VGG [75], ResNet [76], and DenseNet (121, 161) [77]. The findings from the research determined that the DenseNet [77] architecture, specifically DenseNet161 [77], achieves 99.44% accuracy on accepted samples with 31.43% of samples rejected, the best-performing model compared to the other CNN architectures tested in [4]. Different methods for producing metrics to determine when to accept and reject were also investigated in [4]; the best-performing method was ICP-Softmax, which will be utilised in the following investigations.

The first step involved implementing a pipeline using the best-performing architecture, DenseNet161 [77], as the baseline model to create the foundation for further research. This is one of the primary objectives of the research as described in Chapter 1 Section 1.2.1. The research detailed in [4] does not include any specific pre-processing strategies or hyperparameters used. Therefore, the Literature Review will inform the approach for deciding the hyperparameters and pre-processing approach, and additional experimentation will be conducted to determine the most effective model configuration settings. Additionally, the investigation will evaluate the impact of data augmentation, a secondary research objective shown in Chapter 1 Section

1.2.2. The alterations to pre-processing and hyperparameters will be investigated by updating the baseline model and comparing the performance results of the different models.

The next step was to investigate one of the research gaps identified in the Literature Review for my rigorous investigation, another primary objective highlighted in Chapter 1 Section 1.2.1. For the experiment conducted in [4], the accept/reject process is used once training has converged, with the rejected samples left for classification through other means, such as by a radiologist. However, as the Literature Review shows, this can be a time-consuming, complex process and susceptible to human error [8]. A research gap identified in the Literature Review was not utilising the rejected images within the training process. This can be achieved through hard negative mining [78]. Hard negative mining employs a similar strategy to the accept/reject approach shown in [4]. However, instead of utilising the accept/reject after training has converged, hard negative mining determines the rejected samples, the ”hard negatives”, during training and feeds them back into the training dataset to allow for more exposure to rejected cases in the training process. The mining for hard negatives is completed at the end of each training pass of the training data (epoch) with hard negatives added to the training data for the next epoch in the training process.

3.2 Pipeline

The pipeline requires the raw dataset to be stored in appropriate training, validation and testing directories. The directory structure contained a top-level directory (e.g. Train, Validation, Test) and sub-directories for each class (e.g. ABE, ART, BAS, etc.) Within the class directory, each image is stored with the label attached to the file name (e.g. ABE_image_number). Scripts for moving images into this directory structure and splitting them into train/validate/test splits can be found in the *src* folder in *data_preparation*. See the Usage guide in the *src* directory, *README* for a detailed description of the script, the desired directory structure.

The pipeline begins by completing pre-processing steps such as data cleaning, splitting, and

alterations to the sample image to ensure it is valid for input to the model. This is a transfer learning pipeline where the model is pre-trained on ImageNet [43], and given ImageNet [43] weights are invoked to the model before training and tuning on the BMCMHF dataset [3].

Data augmentation is implemented through two approaches: pre-processing and on-the-fly data augmentation. Pre-processing data augmentation uses the training data to create more images by randomly altering the images in classes via rotations, horizontal flips and colour alterations as shown in Figure 3.1. On-the-fly data uses the augmentation probability rate described in Figure 3.2 to augment images randomly as they are loaded to be used, altering the image through random rotation or colour alterations. On-the-fly augmentation is repeated at each epoch to help ensure that the model does not overfit to the training data. At each epoch the data augmentation is completed using the original training dataset, the previous on-the-fly data augmented training dataset is not re-used at subsequent epochs in training. This allows the model to learn more generalised variations of data. The combination of data augmentation approaches helps ensure the model does not overfit the training data and allows the model to generalise more successfully.

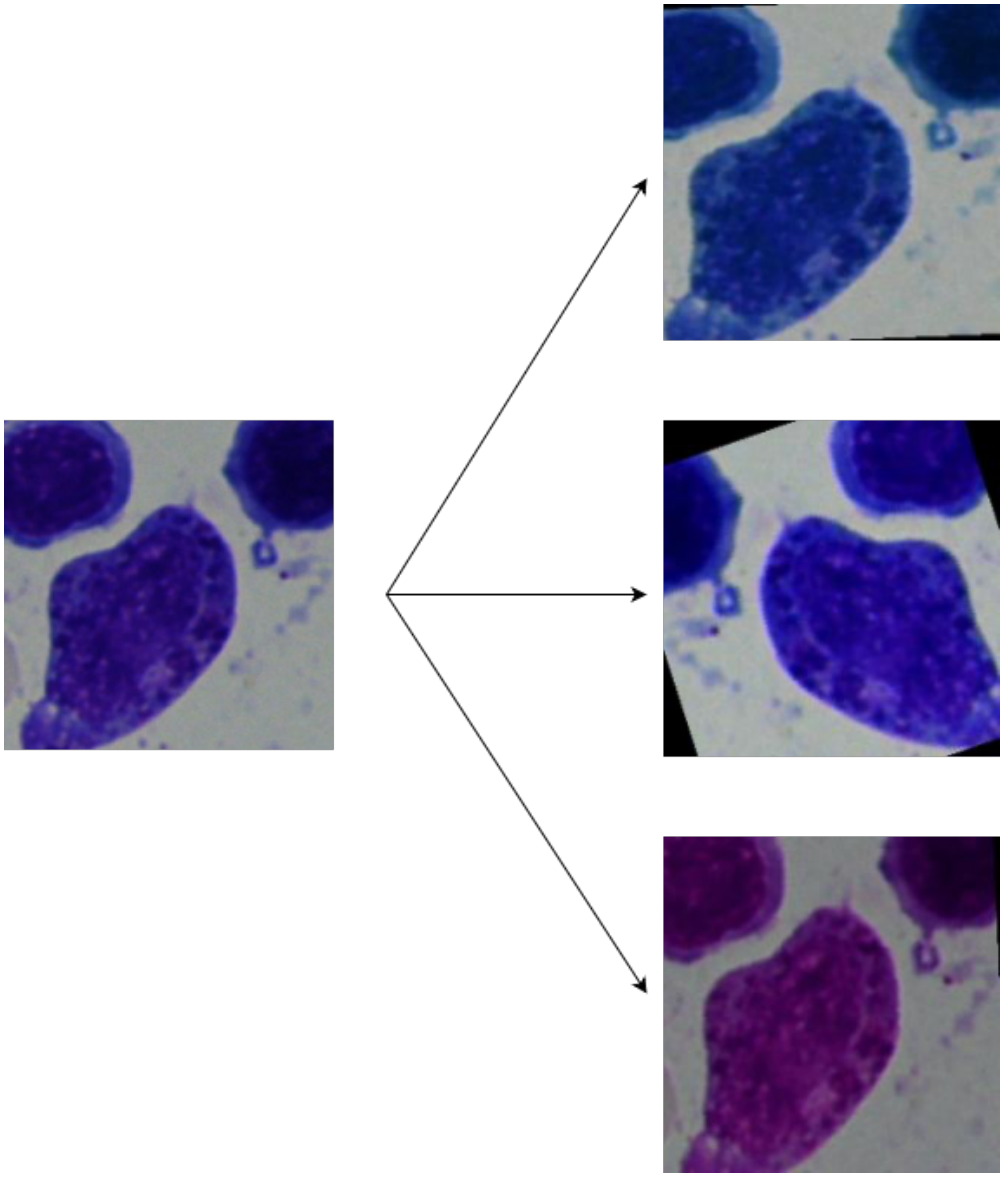


Figure 3.1: *Example of Augmentation Results.*

Alteration	Range	Random/Step
Horizontal Flip	-	Yes
Rotation	0 - 45 (degrees)	1 degree
Colour Jitter	Brightness=0.0-0.2, contrast=0.0-0.2, saturation=0.0-0.2, hue=0.0-0.1	0.01
Rezised Crop	224 (image size), scale to crop (0.8-1.0)	0.01

Figure 3.2: *Available Data Augmentation Alterations.*

The training process has a maximum of 100 epochs, iterations through the training data, which can be performed during training. After each epoch finishes, the accuracy of the validation set

is calculated. If the validation accuracy does not improve for 10 epochs in a row, the patience value, early stopping is invoked, and training has converged. A consistent downward trajectory of the validaiton accuracy is a sign that the model is beginning to overfit [79], early stopping helps to mitigate that problem.

Once the training has converged, the calibration set will be utilised. In the pipeline, this is a portion of the training data that is separated before training begins. The calibration set comprises 10% of the training set, split randomly across the data. The purpose of the calibration set is to determine the best possible confidence threshold value to yield the highest accuracy value in evaluation. The confidence threshold value is used once training has converged to accept/reject samples before evaluation. The calibration set provides an unbiased sample of data to evaluate the performance using differing confidence threshold values. The threshold values range from 0.00-0.95 and are stepped along in 0.05 increments; at each threshold value, the accuracy of predictions is calculated. The best threshold is determined by the threshold that generates the highest accuracy.

Similarly to the pipeline [4], the accept/reject mechanism uses a softmax function to determine the confidence threshold value. A softmax function is commonly used in classification tasks; it outputs a probability distribution from 0 to 1 for all classes, summing to 1 [80]. For each class a probability that the prediciton is that class is generated. This can determine the likelihood that a prediction about a class is confident. In this context, a prediction is determined to be confident if the highest probability value for the predicted class is above a certain confidence threshold. In the accept/reject mechanism, if a prediction of a sample is confident, it is accepted; otherwise, that sample is rejected. In [4] if a sample is accepted it is used for evaluation on the validation and test set, otherwise in practice it would be left to be analysed in some other way, by a professional manually for example.

The baseline model for this investigation will implement the same accept/reject strategy as [4]

by accepting and rejecting samples after training has converged using a softmax function to calculate the highest probability in the distribution of probabilities across all classes.

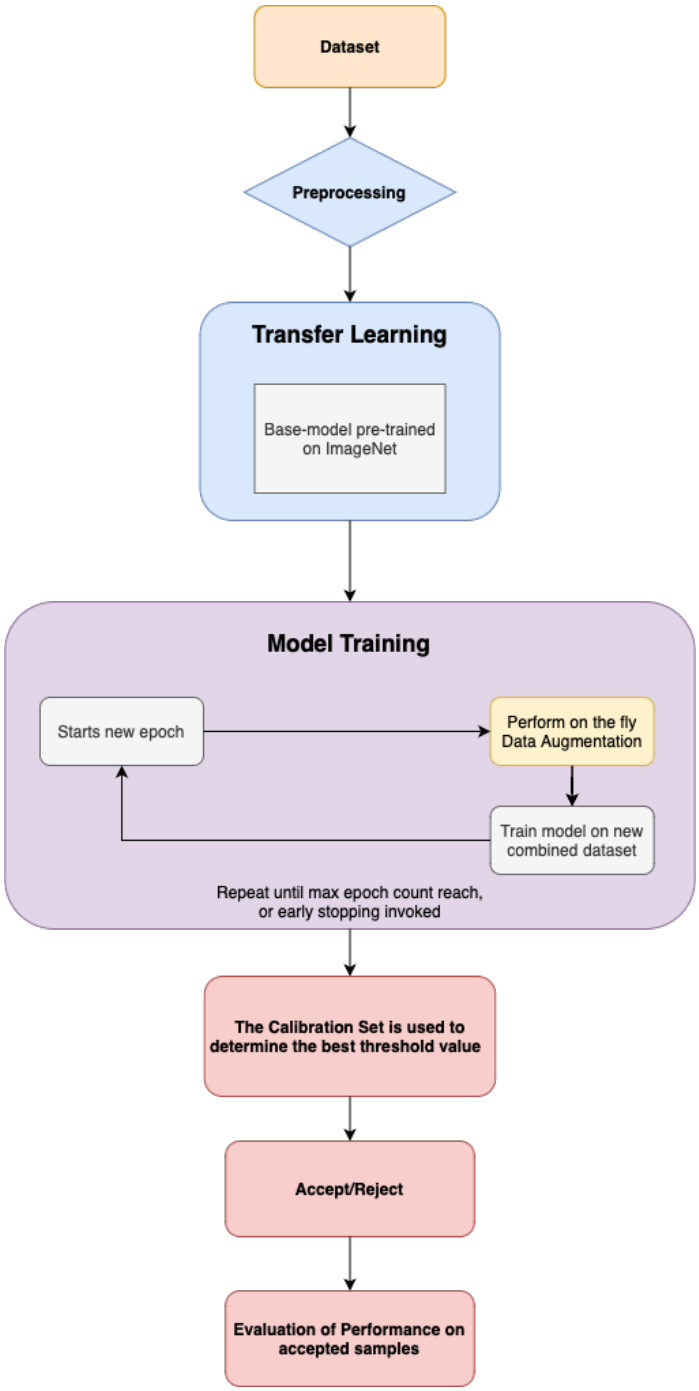


Figure 3.3: *Flowchart of pipeline with Accept/Reject.*

The hard negative mining uses a confidence threshold detailed in Table 3.1 and after the model makes predictions, it uses a confidence threshold calculation to determine any predictions

where the confidence value is below the confidence threshold value, making it a hard negative for the model. These are collated and added to the training data once the next epoch begins. This allows the model, when training, to get more exposure to instances where it has less confidence regarding the prediction made.

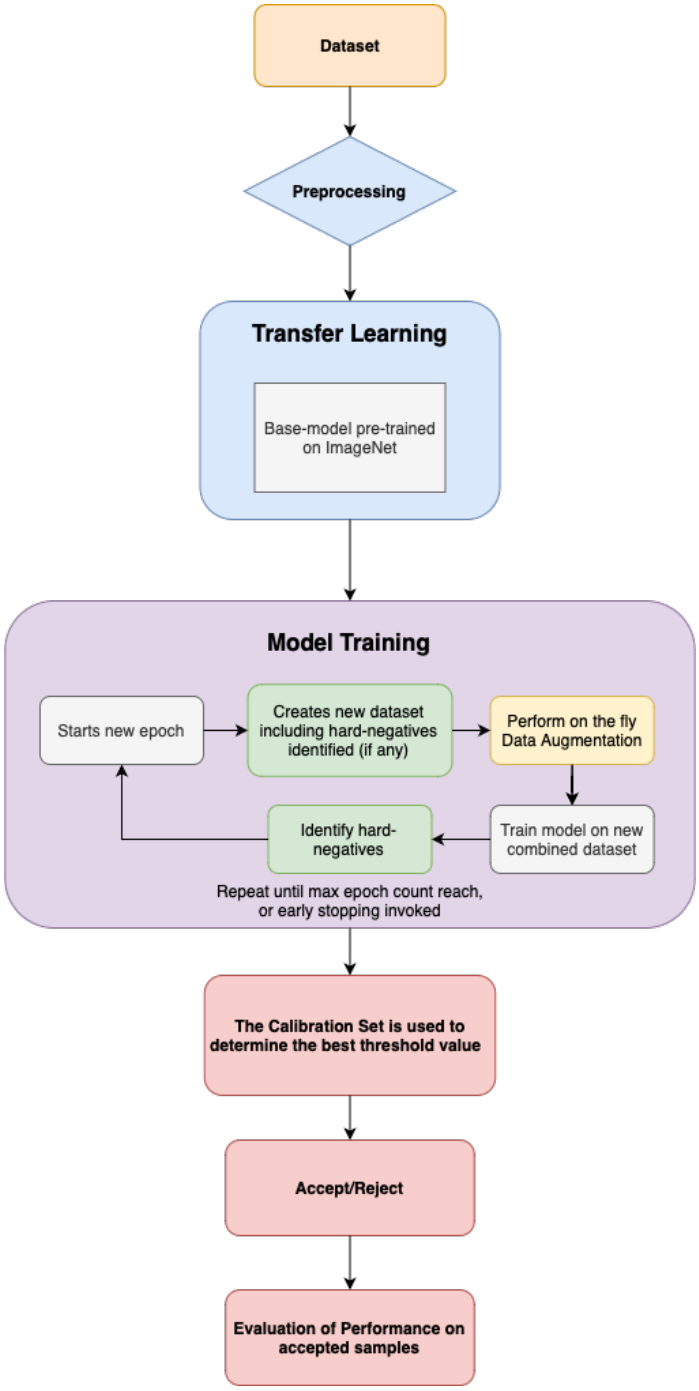


Figure 3.4: *Flowchart of pipeline with Hard Negative Mining.*

3.3 Dataset

The BMCMHF dataset [3] was split into train and test with 80% allocated to train and 20% for test, the validation dataset was generated using 20% of the train dataset. This split was opted for due to some classes having low amounts of data available, ensuring that the test results per class are robust. Further the calibration set was made from 10% of the training dataset, to use to calibrate the confidence threshold value after training converges.

3.3.1 Data Augmentation

Data augmentation is a common technique for dealing with overfitting in training. The baseline model used data augmentation; however, this section will implement a more aggressive data augmentation approach to help mitigate the rate of overfitting. The new approach will increase the data augmentation images created in pre-processing and a larger on-the-fly augmentation probability when loading the training data. This approach will generate a larger variation of images the model uses for training, preventing the model from learning complex patterns within the training data. The unaugmented dataset [3] can be seen in Figure 3.5. After augmentation, the dataset has a more balanced overall representation, as shown in Figure 3.6.

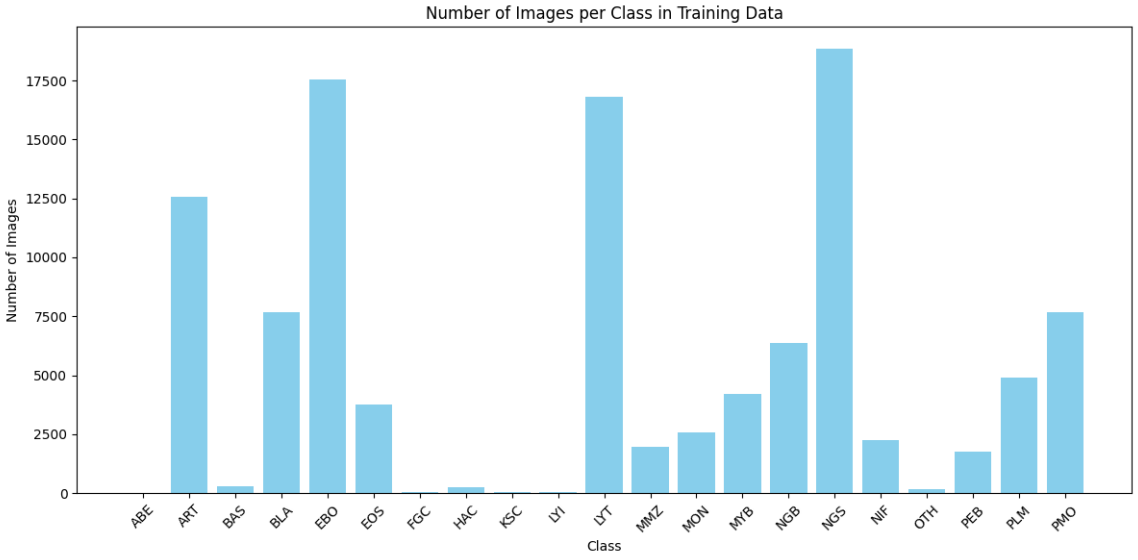


Figure 3.5: *Dataset [3] Distribution of Classes in Training Set.*

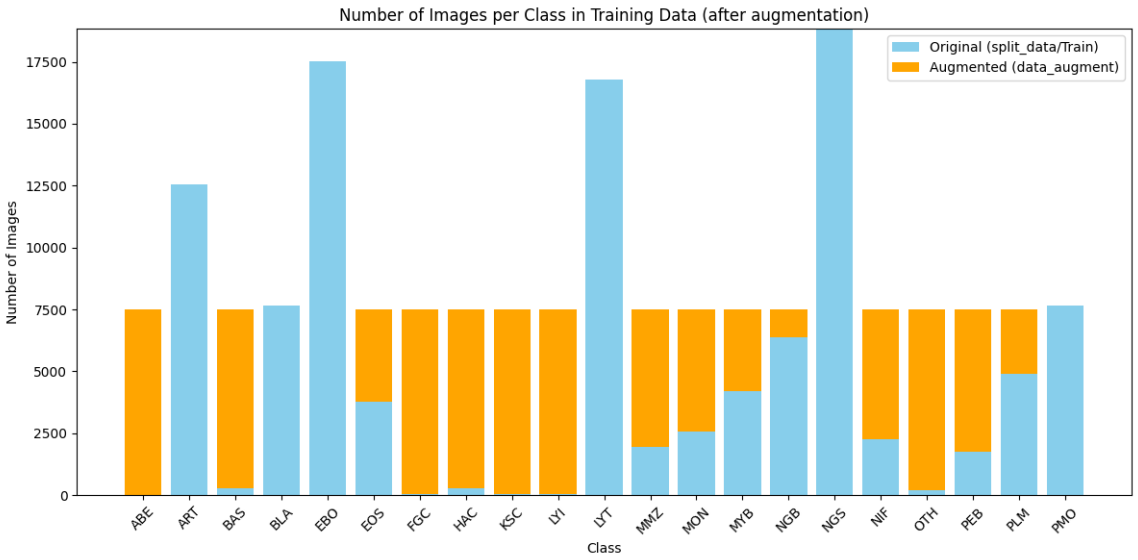


Figure 3.6: *Dataset [3] Distribution of Classes in Training Set After New Data Augmentation.*

The data augmentation strategy ensured that all classes in the training set had almost as many samples as the 5th most common class in the dataset, BLA. Each class containing less than 7,500 samples had data augmentation performed equally on each available sample image until the class contained 7,500 samples. This approach generated the training set shown in Figure 4.6. The reason for this decision was that generating further samples to ensure a higher amount of samples than 7,500 per class would encourage overfitting within classes, even though the augmentation is randomised and uses a wider variation of alterations if the amount of samples available before augmentation is low many samples when augmented will generate a similar result that could lead to the updated baseline model memorising the samples rather than learning the general patterns within the class samples.

3.4 Technology

To develop the pipeline: PyTorch was utilised in Python for handling the large datasets and implementation of the CNN models. The code was developed and run in a GPU-accelerated environment, supporting the most recent version of PyTorch at the time of writing (2.6).

3.5 Performance Metrics

During the investigation, the test set was hidden completely and the validation set was used to gauge the performance of the models. At the end of training the model was evaluated on the validation set to produce performance metrics: accuracy, balanced accuracy, precision, recall and f1-score. These metrics were implemented using Sci-kit learn [81] and were used to create graphs to highlight the models performance. A confusion matrix was also generated to show the class balance of predictions.

In the following equations: TP represents True Positive, TN is True Negative, FP is False Positive and FN is False Negative.

3.5.1 Accuracy

The accuracy metric measures the ratio of correct predictions over the total number of instances that are evaluated. Overall, how many predictions that are made are correct [82].

$$Accuracy = \frac{(TP+TN)}{(TP+FP+TN+FN)}$$

3.5.2 Balanced Accuracy

The balanced accuracy (weighted) metric takes into account any class imbalance present [83], ensuring each class contributes no matter equally no matter the class distribution.

$$BalancedAccuracy = \frac{1}{K}\sum_{k=1}^{K}\frac{TP_k}{TP_k+FN_k}$$

TP_k represents the number of True Positives for class k, FN_k represents the number of False Negatives for class k, and K represents the total number of classes present in the dataset.

3.5.3 Precision

The precision metric is used to measure the ratio of positive predictions made that are truly positive [82].

$$Precision = \frac{TP}{(TP + FP)}$$

3.5.4 Recall

The recall metric, also known as sensitivity, measures the ability to identify all relevant positive instances, a higher recall means that fewer truly positive instances are missed [82].

$$Recall = \frac{TP}{(TP + TN)}$$

3.5.5 F1-Score

The F1-Score gives the harmonic mean of precision and recall. This metric aims to be high when precision and recall are both high. It is useful to find the balance between precision and recall, labelled as the F-Measure in [82].

$$F1 = 2 * \frac{(Precision * Recall)}{(Precision + Recall)}$$

3.6 Hyperparameter Choices

Hyperparameters are a model's settings and configuration; altering the parameters changes how a model learns. This section focuses on the hyperparameters and settings: epochs, batch size, learning rate, optimiser, β_1, β_2, weight decay, dropout layer, confidence threshold value, on-the-fly data augmentation probability, the type of loss function, patience, pre-trained and Max. hard negatives.

Epochs are the number of iterations through the entire training data during training [84]. The

batch size determines the number of samples in a batch. A model is trained by iterating through batches; after each iteration, the internal weights of the model are updated. The learning rate determines the strength of newly learned information to override old learned information [85]. The learning rate is between 0 and 1, where a factor of 0 means the model will not learn anything, and a factor of 1 will make the model only consider the most recent information [85]. An optimiser is used during training to adjust the model parameters/weights to improve performance. For this investigation, two variations of optimisers will be used Adam (Adaptive Movement Estimation) [86] and AdamW. β_1 and β_2 control the exponential decay rates for the first and second moment estimates in the Adam optimisation algorithm. They are crucial in determining the smoothness of the parameter updates and are typically set to values close to 1 (e.g. $\beta_1 = 0.9$, $\beta_2 = 0.999$) to ensure stable convergence [86]. Weight decay is a regularisation strategy that penalises large weights during training, effectively adding a term to the loss function. This hyperparameter helps mitigate overfitting by encouraging the network to maintain smaller weight values, thus promoting generalisation. The dropout layer is implemented to randomly deactivate a proportion of neurons during each training iteration. This stochastic approach prevents over-reliance on any single feature, thereby reducing the likelihood of overfitting and enhancing the model's robustness. The confidence threshold sets a minimum probability level that a prediction must exceed to be considered valid. It plays a critical role in decision-making processes for classification and detection tasks, helping to balance precision and recall by filtering out low-confidence predictions. On-the-fly data augmentation introduces random transformations to the training data during the learning process. The augmentation probability specifies the likelihood that any given input sample will be augmented, thereby enriching the diversity of the training data and improving the model's generalisation capability. The loss function quantifies the error between the predicted outputs and the true labels. Selecting an appropriate loss function is imperative, as it directly influences the optimisation process. Early stopping ensures that the model converges before overfitting can occur. The process uses the validation accuracy metric to determine when overfitting occurs. If the validation accuracy value does not improve for a certain number (patience) of epochs in a row, the training converges, and the accept/reject mechanism occurs to evaluate the model's performance.

Pre-trained weights involve initialising a model with parameters derived from a network that has been previously trained on a related task. This transfer learning approach can significantly accelerate convergence and improve performance, particularly when the available training data is limited. The maximum number of hard negatives is a hyperparameter that limits the number of difficult negative samples included during training. Hard negatives—instances that are challenging for the model to classify correctly—are instrumental in refining the model's decision boundary, yet must be controlled to avoid destabilising the learning process.

Model	Hyperparameter	Value	Justification
Baseline & Updated Baseline	Max No. Epochs, allows for early stopping	100	Ensure enough epochs for the training process to converge while allowing early stopping to prevent overfitting.
Baseline & Updated Baseline	Batch Size	64	A standard batching size for images of size 224x224, balancing computational efficiency and model performance.
Baseline & Updated Baseline	Learning Rate	0.0001	A low learning rate is selected to ensure incremental adjustments to the model weights, promoting stable convergence and reducing the risk of overshooting minima [85].
Baseline & Updated Baseline	β_1	0.9	Standard choice in the Adam optimisation algorithm due to both theoretical reasoning and extensive empirical validation [87].
Baseline & Updated Baseline	β_2	0.999	Standard choice in the Adam optimisation algorithm due to both theoretical reasoning and extensive empirical validation [87].
Updated Baseline	Weight decay	0.01	Weight decay is applied as a regularisation measure to penalise large weights, thereby reducing the risk of overfitting and enhancing generalisation.
Baseline	Optimiser	Adam	From literature [88] it was shown that the Adam optimiser consistently outperformed other options during the experimental testing phase.
Updated Baseline	Optimiser	AdamW	A variant of Adam, the AdamW optimiser, is chosen for the Updated Baseline to investigate the impact of more effective regularisation on the training dynamics.
Updated Baseline	Dropout	0.25	A dropout rate of 0.25 is employed to prevent overfitting by randomly deactivating a fraction of neurons during training, thereby encouraging robust feature learning.
Baseline & Updated Baseline (when hard negative mining is utilised)	Confidence Threshold value	0.65	This threshold is chosen to filter out low-confidence predictions during inference, balancing precision and recall, particularly in hard negative mining scenarios.
Baseline	On-the-fly data augmentation probability	0.5	A moderate augmentation probability of 0.5 introduces sufficient variability in the training data without excessively altering the underlying distribution, thus enhancing generalisation.
Updated Baseline	On-the-fly data augmentation probability	0.7	An increased augmentation probability of 0.7 is adopted to further enrich the diversity of training samples, thereby bolstering model robustness and reducing overfitting.
Baseline & Updated Baseline	On-the-fly data augmentation types	Shown in Figure 3.2.	A various selection of standard augmentation alterations.
Baseline & Updated Baseline	Loss function	Cross Entropy Loss	To replicate [4] and due to its common use in image classification tasks, cross entropy loss is employed for its effectiveness in measuring prediction errors.
Baseline & Updated Baseline	Pre-trained	ImageNet	Pre-trained weights from ImageNet are used to leverage transfer learning, accelerating convergence and improving performance, particularly with limited data [4].
Baseline & Updated Baseline	Patience	10 epochs	The patience parameter is set to 10 epochs, providing the model with sufficient opportunity to improve before early stopping is invoked to prevent overfitting.
Baseline & Updated Baseline	Max. hard negatives	3500	Limiting the maximum number of hard negatives to 3500 maintains a balanced representation of challenging negative samples without overwhelming the learning process, thereby ensuring training stability.

Table 3.1: *Hyperparameter choices for Baseline & Updated Baseline models.*

CHAPTER 4

Validation Set Investigation

This section will describe the investigations carried out on the validation dataset, followed by an outline of the findings. The investigations will include the findings when replicating the deep learning pipeline shown in [4]. This paper implemented a pipeline that uses an accept/reject mechanism after the training has converged. When completing the evaluation of the validation set, a confidence threshold is used to reject samples where the confidence in a correct prediction is below the accepted rate, also called the confidence threshold. Section 4.1 will highlight the findings of replicating this pipeline detailed in [4] and is referred to as the ”baseline model” hereafter. Further investigations will examine the impact of altering the baseline models' hyperparameters and pre-processing teechniques. This ”updated baseline model” will implement a more aggressive data augmentation strategy and change to hyperparameters such as the weighting decay and learning rate to improve models' performance and reduce overfitting to the training data. Section 4.2 will detail the findings of the updated baseline model.

A primary objective of this research is to complete a rigorous investigation of an identified research gap informed by the Literature Review. The baseline model, which replicates the model defined in [4], performs the accept/reject after the training has converged, and in practice, the rejected samples would be analysed by a radiologist. A research gap identified in the Literature Review was to perform the accept/reject process within the training process by implementing hard negative mining. Utilising this technique would allow the hard negative samples to be fed back into the training data. This is an important area to research as it allows the model to

learn from hard negative samples more frequently in training, this approach has been shown to improve model performance. Further, an approach that reduces the amount of rejected samples would, in practice, lessen the workload on professionals like radiologists after the accept/reject process is completed. To determine the effectiveness of this technique, the baseline and updated baseline models will be used with hard negative mining and compared to findings in Sections 4.1 and 4.2, with the findings from implementing hard negative mining shown in Section 4.3.

Section 4.4 shows the best results from investigations using the validation set.

4.1 Baseline Model

The baseline model will implement the basic CNN pipeline architecture shown in [4] and will be used as a baseline for further investigation. The baseline model utilised a CNN-based architecture DenseNet-121 [77] model using transfer learning with pre-trained ImageNet [43] weights to replicate the [4] pipeline. The DenseNet-161 [77] model was selected due to high-performance metrics reported in [4]. After the training process has converged, the accept/reject mechanism is enacted, with accepted samples used to evaluate the model performance and rejected samples discarded. The accept/reject confidence threshold determines whether samples should be accepted or rejected. After the training has converged, a calibration set, a small portion of the data set aside from training and validation, is used to determine the most effective confidence threshold value for that model that yields the highest validation accuracy on accepted samples. The confidence threshold value is in the range 0.00 - 0.95, and for calibration, it is stepped at 0.05 intervals to determine the threshold that results in the highest accuracy value.

The baseline model was trained on the training data and evaluated using the validation set. Minimal explicit hyperparameters were mentioned in [4]. Therefore, the Literature Review and hyperparameter tuning informed the baseline model hyperparameter choices. See Table

3.1 for more details regarding the justification for hyperparameter and model configuration choices. The baseline model has hyperparameters: learning rate = 0.00001, batch size = 32, max epochs = 100 with early stopping allowed, image size = 214x214, pre-trained weights on ImageNet [43]. The baseline model used pre-processing techniques such as data augmentation on the training data and image re-sizing to ensure the image samples entering the model are the correct size. Additionally, any corrupted samples are identified and removed before training begins. During training, on-the-fly augmentation is completed each time the training data is loaded. As each sample is loaded, there is a probability of 0.5 that it will be randomly augmented to help the model generalise and not overfit to the training data.

Figure 4.1 demonstrates the accuracy of predictions made on the training and validation sets during training. The accuracy curves follow the typical trend seen in model training; the training accuracy consistently outperforms the validation accuracy across epochs. This is because the training accuracy uses samples it is learning from to evaluate, whereas the validation accuracy is being performed on samples the model has not been trained on.

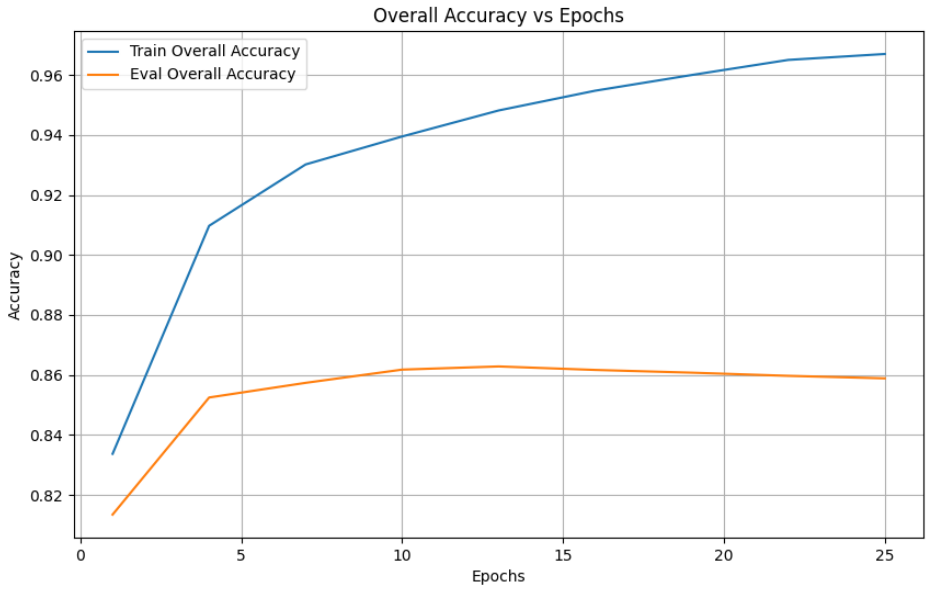


Figure 4.1: *Baseline Model: Accuracy vs Epochs.*

The model's early stopping is invoked at epoch 25 due to no improvement in the evaluation

accuracy. A continued increase in training accuracy whilst validation accuracy decreases is a sign of overfitting in the model [70]. Stopping the model early ensures that when performing evaluation on the model, it reduces the amount of overfitting allowed.

After the training process converges, the accept/reject process is completed. The confidence threshold value is determined using the calibration set, which is set aside from training and validation. The confidence threshold value that produces the best accuracy value is used as the threshold value for accepting and rejecting samples when evaluating. Confidence threshold values from 0.00 - 0.95 are looped through in 0.05 steps to determine the highest accuracy value. The calibration set determines the confidence threshold that yields the greatest validation accuracy value.

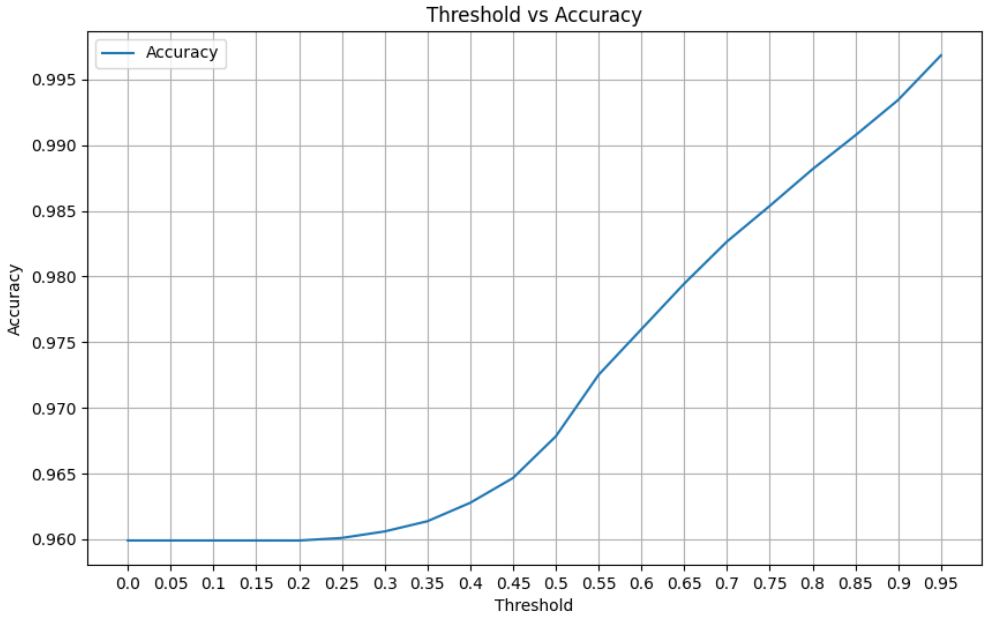


Figure 4.2: *Baseline Model: Accuracy on Accepted Samples vs Confidence Threshold after Training Converges.*

Figure 4.2 demonstrates the accuracy value when evaluating the calibration set for different confidence thresholds. It can be seen that the highest threshold value available when calibrating, 0.95, produces the highest accuracy value. The trend of increasing accuracy as the threshold increases, observed in Figure 4.2, is to be expected. As the threshold increases, more

confidence is required to accept samples, meaning a larger proportion of samples will be rejected, resulting in a higher accuracy value.

The effectiveness of the rejection option can be seen by comparing the performance metrics with and without utilising the accept/reject process. In Figure 4.3, it can be observed that utilising the rejection option greatly improves the performance metrics using the validation set.

Model	**Accuracy**	**Recall**	**Precision**	**F1-Score**	**Rejection Rate**	**Epoch stopped**
Baseline Model	0.941	0.941	0.940	0.940	0.240	25
Baseline Model (without rejecting)	0.858	0.858	0.857	0.856	0.00	25

Figure 4.3: *Baseline Model With and Without Reject Option: Results on validation set after performing Accept/Reject.*

Figure 4.4 shows the performance metric results for the baseline model on accepted samples. The threshold value determined using the calibration set was 0.95, which resulted in the rejection of 24% of the validation samples.

Model	**Accuracy**	**Recall**	**Precision**	**F1-Score**	**Rejection Rate**	**Epoch stopped**
Baseline Model	0.941	0.941	0.940	0.940	0.240	25
Pipeline [4]	0.9944	0.9917	0.9921	0.9920	0.3144	-

Figure 4.4: *Baseline Model and Pipeline [4]: Results on validation set after performing Accept/Reject.*

Comparatively, the [4] pipeline produces a higher accuracy value of 99.44% using a DenseNet161 architecture with an ICP Softmax method for the accept/reject mechanism. The same model

and a similar method for accepting and rejecting were implemented in the baseline model. However, the rejection rate reported in [4] for the same implementation is 31.43% compared to 24% in the baseline model. The dataset used in [4] has 11 classes of 15,998 overall samples compared to the dataset used in this investigation [3], which consists of 21 classes of over 170,000 overall samples. The main difference between the datasets is the imbalance in the number of samples within each class. The difference between datasets can be seen in Figure 4.5 and Figure 4.6, showing the distribution of samples in each class. Figure 4.5 shows the distribution for the dataset used in this investigation [3], and Figure 4.6 shows the dataset distribution for [4].

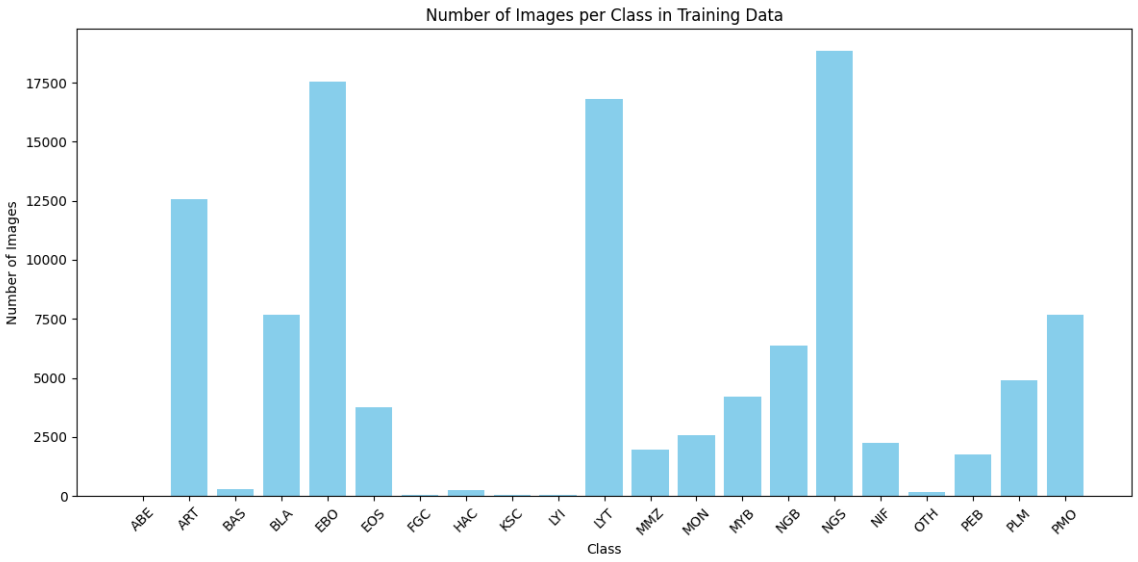


Figure 4.5: *Dataset [3] Distribution of Classes in Training Set.*

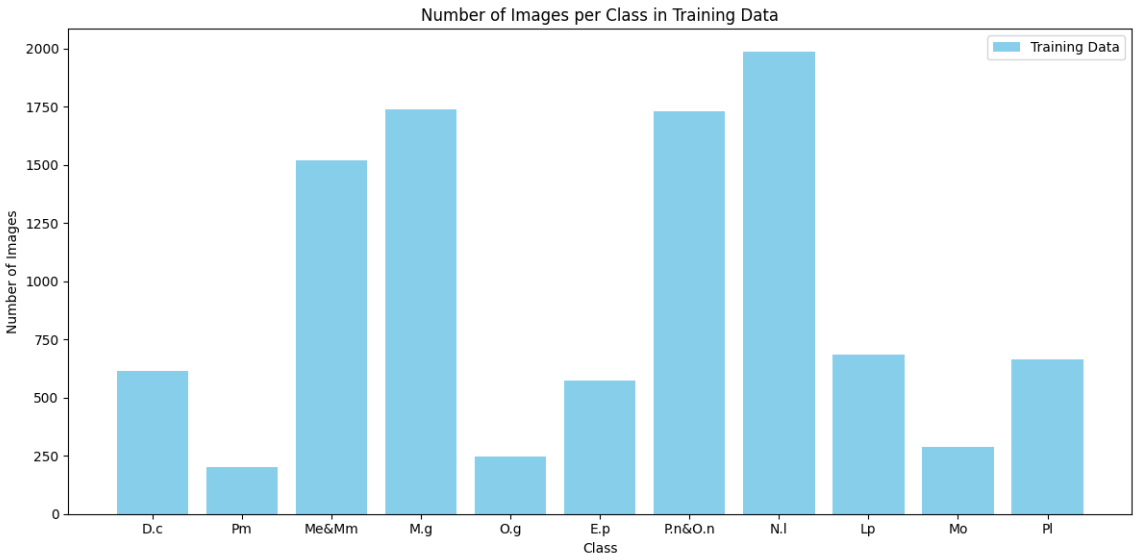


Figure 4.6: *Dataset [4] Distribution of Classes in Training Set.*

Both dataset distributions for [4] and [3] are unbalanced. However, the dataset [3] is significantly more unbalanced. The difference between the class with the most and least samples is over 17,000 for [3], whereas this difference for [4] is around 1,800. As detailed in the Literature Review, the unbalance of classes in the training set has a large impact on the performance achieved. The Updated Baseline Model implemented in Section 4.2 aims to utilise a more aggressive data augmentation approach to help mitigate the impact of an imbalanced training set from [3].

4.2 Updated Baseline Model

This section will investigate the impact of updating the baseline model's hyperparameters and pre-processing strategies. The altered model will be called the "updated baseline model." The updated baseline model will implement a more aggressive approach to data augmentation, further re-balancing the dataset. Additionally, L2 regularisation is added through a weight decay to help to prevent overfitting by penalising large weights and encouraging the model to learn simpler and more general patterns; a dropout layer is placed before the final layer, with a dropout value = 0.25, meaning that 25% of the features learned from training will be set to zero before the final layer. This helps to prevent overfitting and encourages the model to learn generalised patterns within the dataset.

4.2.1 Dropout Layer

The updated baseline model includes a dropout layer before the final layer, which outputs the raw scores for each class. Adding a dropout layer means that a percentage, 25% for the updated baseline model, of the final layer is randomly assigned to zero each epoch. The purpose of a dropout layer is to help prevent overfitting as specific features cannot be relied upon. It also enforces learning on redundant representations, often resulting in better generalisation of unseen data.

4.2.2 Weight Decay

L2 regularisation is implemented through a weight decay set to 0.001 for the updated baseline model. Weight decay helps prevent overfitting by penalising large weights and encouraging the model to learn simpler and more general patterns without fixating on complex characteristics in the training data.

4.2.3 Results

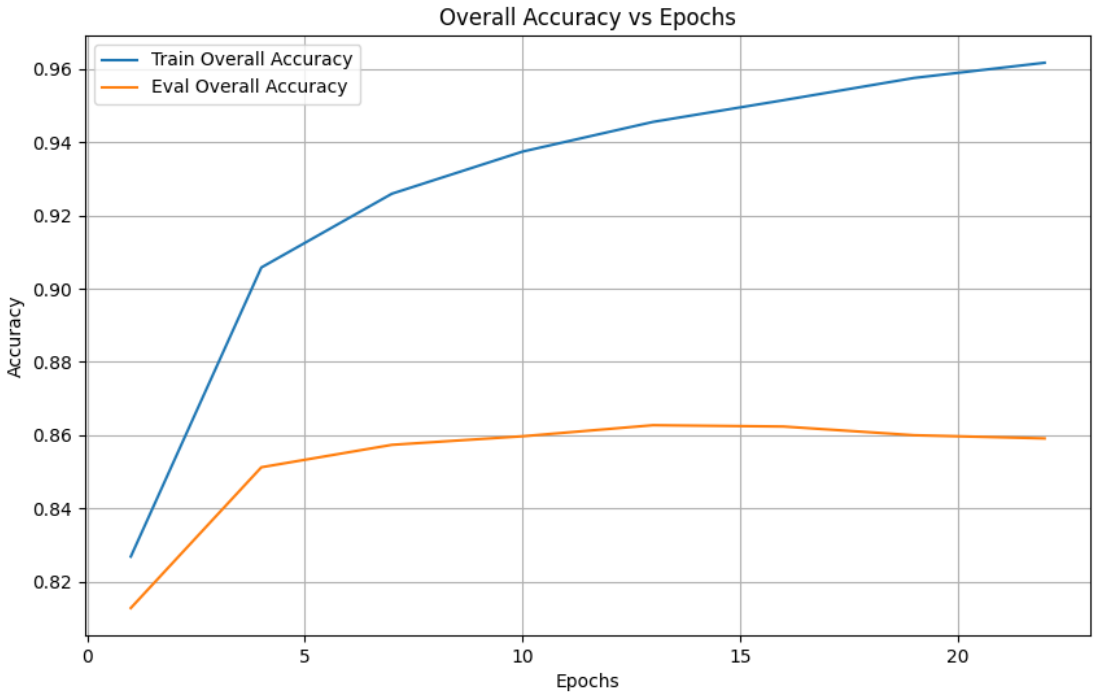


Figure 4.7: *Updated Baseline Model with updated hyperparameters and pre-processing: more aggressive data augmentation approach, AdamW optimiser, weight decay=0.001, dropout rate = 0.25, learning rate = 0.00001, B_1 = 0.9, B_2 = 0.999, On-the-fly augmentation probability = 0.7, max epochs = 100 with early stopping allowed.*

Figure 4.7 shows the updated baseline model's accuracy on the training and validation sets as a function of epochs, with updated hyperparameters and pre-processing approach: weight decay = 0.001, dropout rate = 0.25, learning rate = 0.00001, B_1 = 0.9, B_2 = 0.999, On-the-fly augmentation probability = 0.7.

Figure 4.7 represents the expected trend of increasing model accuracy as training data exposure increases. The training curve consistently outperforms the validation curve. This outcome is expected as the training accuracy value is generated by evaluation performed on the training data, which is the data it has seen. Validation accuracy is generated by evaluation performed on the validation data, which it has not seen and is not trained on. This results in the disparity between the training and validation accuracy in Figure 4.7.

The training and validation accuracy learning curves for the baseline (in Figure 4.1) and updated baseline model (in Figure 4.7) are very comparable, with minimal impact to the accuracy results being observed during the training process.

The updated baseline model also allows for a maximum of 100 epochs to be performed with early stopping, using a metric called patience. Early stopping ensures that the model converges before overfitting can occur. For the updated baseline model, the training process converged at epoch 23.

Model	**Accuracy**	**Recall**	**Precision**	**F1-Score**	**Rejection Rate**	**Epoch stopped**
Baseline Model	0.941	0.941	0.940	0.940	**0.240**	25
Updated Baseline Model	**0.947**	**0.947**	**0.947**	**0.946**	0.270	23

Figure 4.8: *Updated Baseline Model: Results on validation set after performing Accept/Reject.*

The updated baseline model shows improvements in all performance metrics: accuracy, recall, precision, and F1-score compared to the baseline model. Further, the updated baseline model rejects 3% more samples than the baseline model. After the training process converges in both instances, the chosen confidence threshold is 0.95.

An increase in performance metrics for the updated baseline model shows that altering the base model hyperparameters and preprocessing techniques, like the aggressive augmentation technique, has led to more samples being classified correctly. This is down to multiple factors; the training set to learn from is much larger with the updated augmentation strategy implemented. A larger training set results in a more balanced learning process for the model as there is not a large disparity between class sample sizes, and a larger amount of varied samples overall

allows the model to learn features in the data more effectively, as shown by the slight increase in all performance metrics.

4.3 Hard Negative Mining

Hard Negative Mining is a technique that identifies samples during training that are hard negatives and implements them into the training set to give the model more exposure to these examples. From the Literature Review, a research gap was identified in not utilising the accept/reject mechanism within training. In the pipeline [4] that formed the basis for the baseline and updated baseline model, the accept/reject process is completed after training has converged. Performing accept/reject after the training has converged gives no opportunity for the rejected samples to be utilised in the training process. Utilising hard negatives in training encourages the model to learn the samples the model rejects and is not as confident on predicting. Further, in practice, it has been shown from the Literature Review that classifying samples of similar nature by humans is a time-consuming and error-prone process [8]. The aim of implementing hard negative mining is to improve the overall accuracy but also reduce the amount of samples that are rejected, to reduce the workload on professionals to classify the samples that are rejected.

The pipeline implemented previously will be altered to perform hard negative mining. In this investigation, a hard negative is any sample rejected by the model because it does not meet the confidence threshold. After an epoch is completed, the model will perform an evaluation using the validation set. A list of hard negatives, capped at a maximum value, is collated and appended to the training set for the subsequent epoch. This process is shown in Section 3 and Figure 3.4. It is important to note that the new training set only contains the hard negatives identified in the most recent epoch; the hard negative identification process is completed at the end of each epoch. As the model is trained further, its confidence for each sample will alter from epoch to epoch, so only utilising the hard negatives identified in the most recent epoch ensures that the most up-to-date hard negatives are used. A hard negative identified at one epoch

may not be considered a hard negative at another epoch.

To determine the impact of implementing hard negative mining in training, the baseline and updated baseline model will both be extended to implement hard negative mining.

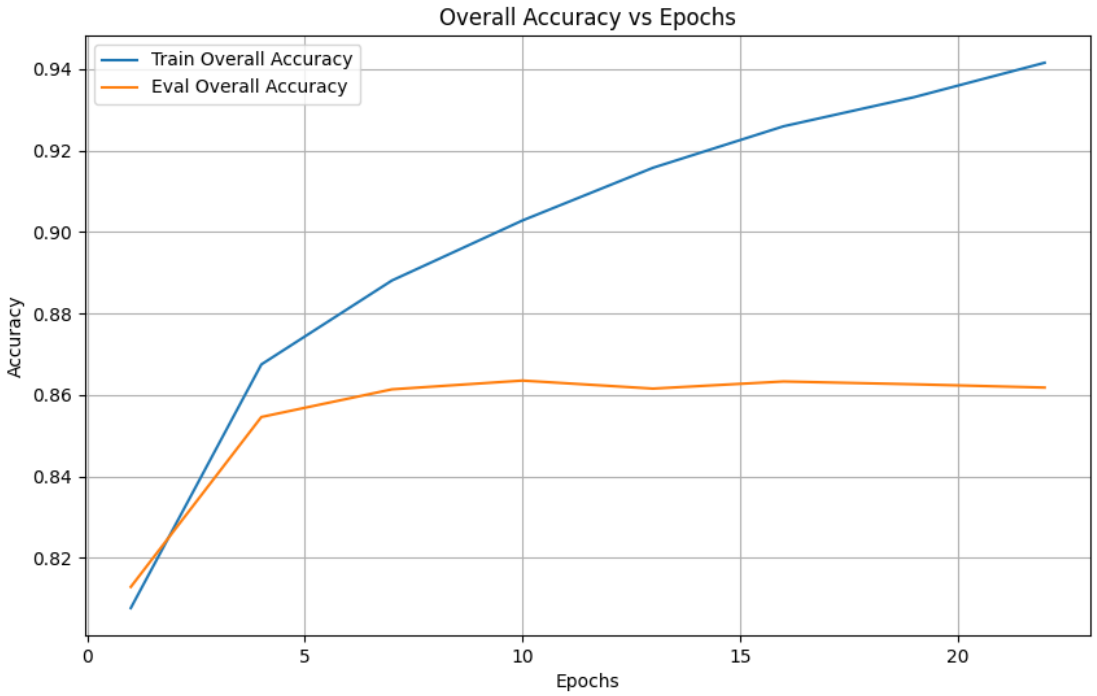


Figure 4.9: *Baseline Model: Train and Validation Accuracy with Hard Negative Mining.*

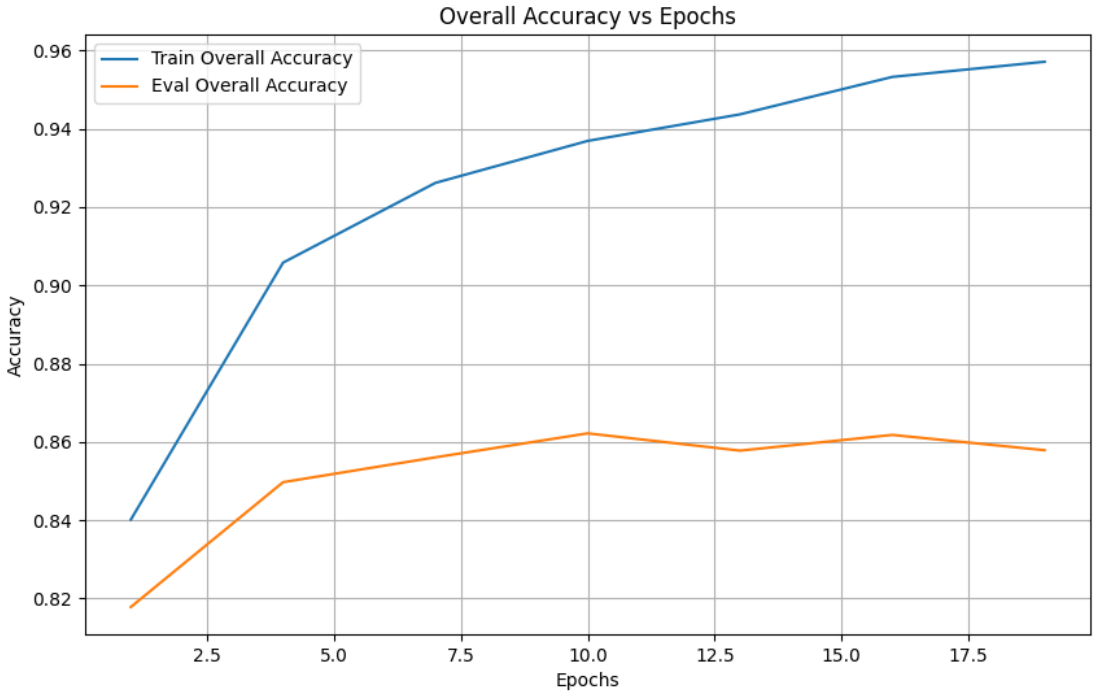


Figure 4.10: *Updated Baseline Model: Train and Validation Accuracy with Hard Negative Mining.*

From Figures 4.9 and 4.10, it can be observed that the training and validation accuracy follows the expected trend of model training with minimal improvements in the earlier epochs compared to Figures 4.1 and 4.7, which do not have hard negative mining implemented. A slightly earlier spike in validation accuracy in Figures 4.9 and 4.10 shows that the model is learning at a slightly faster rate.

The baseline model with hard negative mining training converges at epoch 22; the updated baseline model with hard negative mining converges at epoch 19. The earlier stopping in training further shows that the hard negative mining encourages faster convergence in terms of the number of epochs completed. This is due to adding the hard negatives to the training set, increasing the number of training samples presented to the model each epoch. The training process is also accelerated due to the model having more exposure to hard negative examples, allowing learning on rejected samples to occur earlier.

4.4 Best Validation Set Results

This section details the best results from the investigations using the validation set. As shown in Figure 4.11, the best-performing model is the updated baseline model with hard negative mining implemented. This outperforms each of the other strategies implemented in this investigation.

Figure 4.11 highlights a slight increase in accuracy when utilising the hard negative mining approach for both the baseline and updated baseline models. Further, a more significant discovery is the decrease in the rejection rate when using hard negative mining to 0%, a decrease from 24% and 27% for the baseline and updated baseline models, respectively. Not rejecting any samples means that there is not a large proportion of the evaluated dataset left for analysis by other means, as is the case with the [4] pipeline. As all samples are evaluated when extending the models to utilise hard negative mining, this would eliminate the added workload introduced by rejecting samples. The slight increase in accuracy achieved is promising; however, compared to the [4] pipeline achieving 99.44% accuracy on the validation set, the hard negative mining implementations detailed in this investigation are outperformed.

Model	**Accuracy**	**Recall**	**Precision**	**F1-Score**	**Rejection Rate**	**Epoch stopped**
Baseline Model	0.941	0.941	0.940	0.940	0.240	25
Updated Baseline Model	0.947	0.947	0.946	0.946	0.270	23
Baseline Model: With Hard Negative Mining	0.956	0.956	0.955	0.955	**0.000**	22
Updated Baseline Model: With Hard Negative Mining	**0.958**	**0.958**	**0.957**	**0.957**	**0.000**	19

Figure 4.11: *Results from each of the different investigated models on the Validation Set.*

CHAPTER 5

Test Set Results

The test set is used to evaluate a model's performance on unseen data. This differs from the validation set as alterations are made to the model's hyperparameters and configuration settings to fine-tune the model to the validation set to yield the greatest performance. The test set is used as a final evaluation of the model on unseen data. After the evaluation of the test set was completed, no further alterations to the models configurations were performed.

The test set was used to evaluate the baseline and updated baseline models, both with and without hard negative mining implemented, to further determine the impact of altering the hyperparameters and configuration settings of the model and the effect of using hard negative mining in the training process.

5.1 Baseline Model

Figure 5.1 presents the performance of the baseline model on the test set. The model achieved an accuracy of 94.2%, with recall, precision, and F1-score values all around 94.2%. A rejection rate of 24.4% and early stopping at epoch 25 were observed. These results are consistent with the trends identified during training and validation, confirming that the accept/reject mechanism functions as intended.

Model	Accuracy	Recall	Precision	F1-Score	Rejection Rate	Epoch stopped
Baseline Model	0.942	0.942	0.942	0.941	0.244	25

Figure 5.1: *Baseline Model: Results from Test Set.*

5.2 Updated Baseline Model

hows the test set evaluation for both the baseline and the updated baseline models. The updated baseline model, which utilises a more aggressive data augmentation strategy along with revised hyperparameters (including weight decay and dropout layer), exhibits marginally improved performance. It records an accuracy of 94.7%, with recall and precision matching this value and an F1-score of 94.6%. However, the rejection rate is slightly higher at 27.3%, reflecting a more conservative acceptance criterion from more samples to train on from enhanced data augmentation. This improved performance underlines the positive impact of the modified pre-processing and hyperparameter adjustments on the model's capacity to generalise.

Model	Accuracy	Recall	Precision	F1-Score	Rejection Rate	Epoch stopped
Baseline Model	0.942	0.942	0.942	0.941	0.244	25
Updated Baseline Model	0.947	0.947	0.947	0.946	0.273	23

Figure 5.2: *Baseline and Updated Baseline: Results from Test Set.*

5.3 Hard Negative Mining

The integration of hard negative mining into the training pipeline is intended to further improve the models' robustness by reintroducing challenging samples during training. Figure 5.3 reports the performance of both the baseline and updated baseline models when hard negative mining is implemented. The baseline model with hard negative mining attains an accuracy of

95.2%, while the updated baseline model with hard negative mining reaches 95.5%. Notably, both configurations exhibit a rejection rate of 0%, demonstrating that the models are now capable of confidently classifying all test samples without necessitating any post-hoc rejection. This outcome suggests that hard negative mining effectively enables the model to learn from its challenging predictions during training, thereby enhancing its overall predictive certainty.

Model	**Accuracy**	**Recall**	**Precision**	**F1-Score**	**Rejection Rate**	**Epoch stopped**
Baseline Model	0.941	0.941	0.940	0.940	0.240	25
Updated Baseline Model	0.947	0.947	0.947	0.946	0.273	23
Baseline Model: With Hard Negative Mining	0.952	0.952	0.951	0.951	0.000	22
Updated Baseline Model: With Hard Negative Mining	0.955	0.955	0.954	0.954	0.000	19

Figure 5.3: *Results from each of the different investigated models on the Test Set.*

5.4 Best Test Set Results

Figure 5.4 consolidates the best test set performance metrics across all investigated model configurations. The updated baseline model with hard negative mining achieves the highest performance, with an accuracy of 95.8%, recall of 95.8%, precision of 95.7%, and an F1-score of 95.7%. The complete elimination of the rejection rate underscores the effectiveness of the hard negative mining approach, as it enables the model to confidently classify every sample without recourse to the accept/reject mechanism. These results confirm that the combination of

improved data augmentation, optimised hyperparameters, and the integration of hard negative mining not only enhances the model's performance on the validation set but also translates into superior generalisation on unseen test data.

Model	**Accuracy**	**Recall**	**Precision**	**F1-Score**	**Rejection Rate**	**Epoch stopped**
Baseline Model	0.941	0.941	0.940	0.940	0.240	25
Updated Baseline Model	0.947	0.947	0.947	0.946	0.273	23
Baseline Model: With Hard Negative Mining	0.956	0.956	0.955	0.955	**0.000**	22
Updated Baseline Model: With Hard Negative Mining	**0.958**	**0.958**	**0.957**	**0.957**	**0.000**	19

Figure 5.4: *Results from each of the different investigated models on the Validation Set.*

The test set evaluation confirms that the enhanced configurations—particularly the integration of hard negative mining—yield improved performance relative to the baseline approaches. The updated baseline model with hard negative mining not only delivers higher accuracy and improved precision but also achieves a rejection rate of 0%, demonstrating a significant enhancement in model reliability and reducing the burden of manual review. These findings align with the trends observed in the validation investigations and underscore the benefits of the proposed modifications in achieving superior generalisation on unseen data.

CHAPTER 6

Discussion and Evaluation

In this section, the performance of the developed models is interpreted and evaluated in depth. Three models were implemented and tested on the validation dataset: a baseline model replicating the [4] pipeline, an Updated Baseline Model with improved hyperparameters and data augmentation, and a version of both the models with Hard Negative Mining (HNM) integrated into training. The results are compared to each other and to approaches from the literature including DenseNet121 [77], BoMaCNet [37], and Siamese networks [40]. The significance of these findings is discussed, along with explanations for observed performance differences. Finally, the section addresses the limitations of the current approach – such as class imbalance, generalisation issues, and computational constraints – and proposes future research directions to overcome these challenges and build upon the results.

6.1 Comparison with Literature and Related Work

The performance of the investigated models can be contextualised by comparing it with results reported in the literature for similar bone marrow cell classification tasks. Overall, this investigation determined the best performing model is the Updated Baseline with HNM, that reached 95.8% accuracy on test set, with 0% rejection rate. In the pipeline replicated to provide a baseline model [4], this implementation achieved 99.44% accuracy with 31.44% rejection rate. The choice of using a DenseNet [77] architecture for the baseline was motivated by such findings in literature, and indeed DenseNet [77] proved to be a strong backbone for this task. The gap

between 95.8% and 99.44% suggests there is improvement to be made considering the accuracy of the model, however it is notable that the differences in rejection rate as the replicatd model rejected a much larger amount of samples compared the HNM implementation in this investigation. Reasons for this disparity in accuracy include differences in training data volume as a different dataset was utilised in [4], preprocessing, The achieved 95.8% accuracy aligns with the high performance range of CNNs reported for medical image classification.

Another relevant benchmark is the custom-tailored model BoMaCNet [37]. This CNN was specifically designed for bone marrow cell cytology classification and achieved about 93.06% validation accuracy in their experiments. BoMaCNet's performance set a strong baseline; in fact, it outperformed several tested transfer-learning models in that study. The models exceed this performance level: even the original baseline (94.1% val accuracy) slightly surpasses BoMaCNet, and the updated models reach the 95–96% range. This comparison is encouraging, as it suggests that approach of utilising a more advanced CNN architecute and hard negative mining can outperform the architecture BoMaCNet [37]. It's worth noting, however, that BoMaCNet was limited to classifying six categories of bone marrow cells, the most common types, whereas the in this investigation approach was not limited in this way. The dataset utilised in [37] is the same dataset as in this investigation [3], however limited to the chosen classes.

The literature also documents alternative strategies such as Siamese neural networks and hybrid models. A Siamese network approach [35] on the same dataset as this investigation [3] reached 84% validation accuracy, which is notably lower than the performance of the models in this investigation. The Siamese model's main advantage, as reported, is its ability to work well with limited data by learning similarity metrics between cell image pairs. ndeed, it outperformed simpler approaches like CNN+SVM or CNN+XGBoost combinations which only achieved 28–32% accuracy. For large enough datasets, direct classification with deep CNNs tends to yield higher accuracy than metric-learning approaches. The Siamese network's strength could be relevant if we had very few examples of certain rare cell classes – a scenario where the

model might struggle as indicated by the importance of data augmentation to alleviate class imbalance.

The paper [4] experimented with various CNN architectures (GoogLeNet, VGG, ResNet, DenseNet, etc.) and found that a DenseNet (specifically DenseNet-161) performed best, achieving about 99.44% accuracy on the accepted samples while rejecting 31.43% of the samples. Our baseline implementation did not reach that high accuracy – we saw 94% on validation – which can be attributed to several factors: differences in training hyperparameters and preprocessing ([4] did not detail their exact settings, so the replication had to make assumptions), a different dataset was utilised.

The effectiveness of hard negative mining is evident when comparing to the literature: it allowed us to close the gap in coverage (0% rejects) while still remaining within a few points of the highest reported accuracies. In summary, the findings are in line with the broader literature: deep CNN models (especially DenseNet variants) are extremely powerful for cell image classification, achieving very high accuracy, and techniques like extensive data augmentation and mining of hard examples further improve upon this performance.

6.2 Significance of the Results

The improvements observed with data augmentation and HNM highlight the significance of addressing training data deficiencies. Augmenting the dataset aggressively led to a more robust implementation of the updated baseline model – this underscores the point that in medical imaging, where acquiring large labeled datasets can be difficult, synthetic augmentation of existing data is an effective way to boost performance. The augmentation strategy helped to combat class imbalance (by providing more samples for underrepresented cell types through transformations) and forced the model to learn features that are invariant to slight changes in orientation, scale, lighting, etc. When a model can be trained on varied sample, it is more likely to cope with real-world variability.

The introduction of hard negative mining during training is perhaps the most noteworthy contribution, and its success in the results is significant. It was shown that HNM practically eliminated the need for a reject option by teaching the network to handle those formerly troublesome cases. This suggests that many of the previously "unclassifiable" or low-confidence instances were not inherently impossible to classify – the model just needed additional focused exposure to those patterns. By incorporating HNM, the model's learning became more comprehensive. For the medical domain, this means an AI system can be developed to handle all inputs with confidence, reducing the reliance on human fallback. In a realistic scenario, this could translate to faster diagnoses: if an automated system can classify every cell in a bone marrow sample, a pathologist could receive a full report of cell counts and types, including the presence of abnormal cells, without having to manually review a large subset of ambiguous images. It's important to note that in the experiment, HNM improved accuracy slightly (1-2%). This indicates diminishing returns in pure accuracy from HNM – the baseline was already correctly classifying most easy/medium cases, and the hard cases were a minority of the data. After learning those, the ceiling of accuracy was approached. In comparison, the pipeline [36] which left those cases out achieved a higher accuracy on the accepted samples.

Furthermore, the concept of hard negative mining is widely applicable: it essentially ensures that model training focuses on its weaknesses. In this investigation, the weakness was specific cell images that were confusing (perhaps cells with borderline morphology that overlaps between classes, or rare cell types the model initially had few examples of). By iteratively learning from them, the final model is more expert in those edge cases. This is analogous to how a human trainee might focus study time on the most challenging topics to improve overall competence. The success of HNM in the project hints that similar strategies could be employed in other medical image classification problems.

In diagnosing acute leukemias from bone marrow, pathologists must identify blasts (immature

malignant cells) and count them among other cells. An automated system with the level of accuracy could reliably distinguish blasts from normal lymphocytes, erythroblasts, myelocytes, etc., and provide a count or percentage of blasts with high confidence. If integrated into workflow, this could drastically reduce the time to diagnosis and help standardise results (reducing inter-observer variability inherent in manual counting). The model's high recall means it would catch the vast majority of malignant cells, and high precision means it would rarely misclassify a normal cell as malignant (reducing false alarms). In essence, the effectiveness of CNNs combined with HNM in the study reinforces their potential role as decision support tools in hematopathology. It shows that automation is not only feasible but also reliable for complex tasks like multi-class cell identification, which historically required expert knowledge.

6.3 Limitations

6.3.1 Data Inbalance

The distribution of cell classes in the dataset was imbalanced (some cell types were much rarer than others). This can bias the model to perform better on frequent classes while underperforming on rare classes. Although data augmentation expanded the training set and somewhat alleviated this issue, synthetic transformations cannot fully substitute for real diverse examples of rare cells. The validation metrics we reported are overall averages; it is likely that performance on the least-represented cell categories is lower than the reported average accuracy. Rare but clinically important cell types (for example, blast cells if few in number) might still pose a challenge.

6.3.2 Generalisation to Other Data

The models were trained and validated on a specific dataset of bone marrow cell images. While we took steps to prevent overfitting (augmentation, regularisation), the true test of generalisation is how the model performs on completely independent data (e.g., images from a different hospital, with different staining techniques or microscope settings). There is a risk that the

model has learned dataset-specific nuances and might not generalise well to other settings. For instance, subtle differences in image background, cell presentation, or patient populations could impact performance. Without external validation (beyond the held-out test set from the same source), we cannot be certain the model would maintain 95% accuracy universally. Model generalisability is a known concern in medical AI – models can suffer performance drops when applied in new environments if not carefully retrained or adapted.

6.3.3 Threshold Dependence

The use of a fixed confidence threshold (0.95) for rejecting or accepting predictions is somewhat arbitrary and may not be optimal for all scenarios. While it was chosen based on prior work and yielded a reasonable reject rate (25%), a different threshold would change the balance of accuracy vs. rejection. The HNM process itself depends on this threshold to define "hard negatives." If set too high, many samples get flagged as hard (including some that might be correctly classified but with slightly lower confidence), potentially introducing noise or redundancy in training. If set too low, only extremely misclassified samples are fed back, possibly limiting the benefit. Thus, the effectiveness of HNM and the reported metrics are tied to this threshold choice. A more adaptive approach to confidence or a calibrated uncertainty measure could improve this aspect.

6.3.4 Computational Cost

Training large CNNs like DenseNet on high-resolution cell images is computationally intensive. Incorporating hard negative mining adds to the cost because the model must be evaluated on the validation set at the end of each epoch to harvest hard examples, and the training set is effectively larger in subsequent epochs. In practice, the training times increased due to these additional steps (though the number of epochs to converge decreased). For institutions with limited computing resources, this could be a barrier. Additionally, the final model (DenseNet-161 with thousands of parameters) is relatively heavy; deploying it might require a GPU or a

high-performance CPU to achieve quick inference on hundreds or thousands of cell images. Real-time or near-real-time analysis could be challenging without further optimisation.

6.3.5 Model Interpretability

Like many deep learning models, the classifier operates as a black box with decisions that are not easily interpretable. In medical contexts, the lack of interpretability can be a limitation, as doctors may want to understand why the model classified a cell a certain way (e.g., which visual features were indicative of a blast cell). This investigation did not focus on interpretability methods (such as saliency maps or explainable AI techniques). Without these, it may be harder to trust the model's predictions blindly, especially in borderline cases. This is more of a general limitation of CNNs in medicine, but it applies to this work as well.

6.3.6 Hard Negative Mining Limitations

While HNM was beneficial, its implementation in the project has some constraints. The number of hard negatives added per epoch was capped, and only used the latest epoch's hard negatives (not accumulating all from past epochs). These design decisions, while pragmatic, mean that some hard samples might still not be fully learned if they stop appearing as the "hardest" in later epochs. There is also a possibility that a sample mis-labeled in the dataset (noise in ground truth) could repeatedly appear as a hard negative and the model would be forced to learn a potentially incorrect label, which could mislead training. No observations were made to give any obvious signs of this, but it is a theoretical limitation of feeding back hard negatives without human verification.

6.4 Future Research

Future work should seek to gather a larger and more diverse set of bone marrow cell images, possibly from multiple medical centers. A larger dataset would naturally improve the model's

generalisation. Importantly, efforts should be made to balance the class distribution – for instance, by including more examples of rare cell types (even if it requires targeted data collection from archival samples). In addition to traditional augmentation, more sophisticated methods like synthetic data generation could be explored. Generative adversarial networks (GANs) or other image synthesis techniques might create realistic images of underrepresented cell classes to increase the amount of training data.

While DenseNet proved effective, exploring newer or more powerful architectures could push performance closer to the 98–100% range. Options include Vision Transformers (ViTs) or hybrid models that have shown promise in image classification tasks by capturing long-range dependencies in images. Similarly, trying more recent CNN architectures mentioned in literature might yield gains in accuracy or efficiency. An ensemble of multiple models could also be considered to boost performance, as ensembling often improves robustness and accuracy in image classification.

Future research could experiment with different schedules for integrating hard examples – for example, accumulating a hard example bank over epochs instead of replacing them each time, or dynamically adjusting the confidence threshold as the model improves. Another idea is to incorporate a form of active learning where the model not only learns from its mistakes automatically, but also flags cases where it remains uncertain even after retraining, which could then be reviewed by experts to provide additional insight or labeling. This would merge human-in-the-loop with model-driven hard mining for continuous improvement.

For clinical adoption, it would be beneficial to integrate expert knowledge into the loop. One future direction is to use human-in-the-loop training, where a pathologist can review some of the model's errors or uncertain cases (especially during the early HNM iterations) and provide corrections or insights. This could improve label quality and provide the model with additional context. Moreover, developing explainable AI tools alongside the classifier is important. For

instance, implementing saliency map visualisation (heatmaps on cell images showing which parts the model considered for making its decision) can help verify that the model is looking at biologically relevant features (e.g., nuclear shape, chromatin texture, cell size) when classifying. If explanations are aligned with what a hematologist expects, it increases trust in the model. Future work could also explore case-based explanations – retrieving similar cells from the training set that the model thinks are analogous to the current case – thus providing a form of reasoning.

Finally, expanding the scope beyond just classification of individual cell images could be valuable. Future research could integrate this cell classifier into a larger system that analyses entire bone marrow slides. This might involve object detection to find and crop cells from slide images, then classifying each cell and summarising the overall findings. Such end-to-end automation would be a significant step toward a fully automated bone marrow examination. Additionally, the techniques could be adapted to other related diagnostic areas, such as peripheral blood smear analysis for detecting abnormal blood cells or even other pathology domains. This would test the versatility of the approach and potentially amplify its impact in medical diagnostics.

CHAPTER 7

Conclusion

In conclusion, this dissertation has explored and demonstrated the efficacy of deep learning techniques – in particular, Convolutional Neural Networks – for classifying bone marrow cells in microscopic images, towards the aim of automated blood cancer detection. Several models were developed and evaluated, and a systematic comparison was performed. The baseline model, replicating a known CNN pipeline with an accept/reject threshold, achieved strong initial results (around 94% accuracy on validation) but left a significant fraction of difficult cases unclassified. By introducing a more aggressive training regime with data augmentation and optimised hyperparameters, the updated baseline, achieved higher performance ($\tilde{9}4.7\%$ accuracy) and improved the model's generalisation. The most impactful enhancement was the integration of hard negative mining during training, which led to the best model reaching 95–96% classification accuracy on the validation set while eliminating the need to reject any samples. This model – the updated baseline with HNM – outperformed the other variants in the study, underlining the effectiveness of training the network on its own hard examples. It converged faster and produced balanced high precision and recall, indicating a robust classifier capable of handling the full spectrum of cell images presented.

When comparing the models, it is observed that each improvement (baseline $\rightarrow$ updated $\rightarrow$ HNM) contributed incrementally: the updated baseline's use of extensive augmentation improved performance over the original baseline, and applying HNM further boosted accuracy and confidence. The final model's performance was competitive with, though slightly be-

low, the highest figures reported in literature (e.g., DenseNet161 models with +99% accuracy). However, the approach outlined in this report has the distinct advantage of not requiring a "reject option" – the model provides an outcome for every cell image – which is a valuable property for practical deployment. In essence, the research validated the hypothesis that hard negative mining can enhance CNN training for medical image classification. It closed the loop on the accept/reject paradigm by bringing those previously rejected cases into the training cycle, thereby increasing the model's capabilities. The outcome is a more comprehensive system: one that not only achieves high accuracy on average but also has learned to tackle the most challenging inputs.

The findings of this work confirm that CNNs are a powerful tool in medical image analysis, capable of reaching accuracy levels that approach those of human experts for cell classification. The use of DenseNet architectures provided state-of-the-art feature extraction and classification performance, as evidenced by the results and corroborated by other studies in the field. This underscores the broader implication that deep learning models can reliably perform tasks that were once solely in the domain of specialised medical practitioners. For the specific case of blood cancer detection, having an automated system classify bone marrow cells can significantly expedite the diagnostic process. For example, diagnosing leukemias involves identifying and counting atypical cells; the system could perform this laborious task in a fraction of the time it would take a human, and with consistent accuracy. Such efficiency can lead to earlier detection and treatment, which is critical for patient outcomes in aggressive cancers.

In summary, this dissertation has demonstrated that deep learning techniques, using data augmentation and hard negative mining, are highly effective for classifying bone marrow cells and thereby aiding in blood cancer detection. This investigation achieved high accuracy comparable to state-of-the-art models, while also addressing practical considerations like what to do with low-confidence predictions. The results contribute to the field by showing a viable path to integrate an accept/reject mechanism into training for greater overall benefit. The broader implication is a promising one: with continued research and development, AI systems can become

reliable partners in medical diagnostics, handling complex image classification tasks with accuracy and efficiency. The work presented lays a strong foundation for such future innovations, and emphasises the importance of marrying technical advancements with clinical insights to ultimately improve healthcare outcomes.

UNIVERSITY OF ST ANDREWS
TEACHING AND RESEARCH ETHICS COMMITTEE (UTREC)
SCHOOL OF COMPUTER SCIENCE
PRELIMINARY ETHICS SELF-ASSESSMENT FORM

This Preliminary Ethics Self-Assessment Form is to be conducted by the researcher, and completed in conjunction with the Guidelines for Ethical Research Practice. All staff and students of the School of Computer Science must complete it prior to commencing research.

This Form will act as a formal record of your ethical considerations.
Tick one box

☐ **Staff Project**
☐ **Postgraduate Project**
☒ **Undergraduate Project**

Title of project

Deep Learning Techniques for Blood Cancer Detection by Cell Classification in Bone Marrow Cellular Images

Name of researcher(s)

Lewis Carmichael

Name of supervisor (for student research)

David Harris-Birtill

OVERALL ASSESSMENT (to be signed after questions, overleaf, have been completed)

Self audit has been conducted **YES** ☒ **NO** ☐

There are no ethical issues raised by this project

Signature Student or Researcher

Print Name

Lewis Carmichael

Date

25.09.2024

Signature Lead Researcher or Supervisor

Print Name

David Harris-Birtill

Date

25.09.2024

This form must be date stamped and held in the files of the Lead Researcher or Supervisor. If fieldwork is required, a copy must also be lodged with appropriate Risk Assessment forms. The School Ethics Committee will be responsible for monitoring assessments.

Research with secondary datasets

Please check UTREC guidance on secondary datasets (https://www.st-andrews.ac.uk/research/integrity-ethics/humans/ethical-guidance/secondary-data/ and https://www.st-andrews.ac.uk/research/integrity-ethics/humans/ethical-guidance/confidentiality-data-protection/). Based on the guidance, does your project need ethics approval?

YES ☐ **NO** ☒

** If your research involves secondary datasets, please list them with links in DOER.*

Research with human subjects

Does your research involve collecting personal data on human subjects?

YES ☐ **NO** ☒

If YES, full ethics review required

Does your research involve human subjects or have potential adverse consequences for human welfare and wellbeing?

YES ☐ **NO** ☒

If YES, full ethics review required
For example:
Will you be surveying, observing or interviewing human subjects?
Does your research have the potential to have a significant negative effect on people in the study area?

Potential physical or psychological harm, discomfort or stress

Are there any foreseeable risks to the researcher, or to any participants in this research?

YES ☐ **NO** ☒

If YES, full ethics review required
For example:
Is there any potential that there could be physical harm for anyone involved in the research?
Is there any potential for psychological harm, discomfort or stress for anyone involved in the research?

Conflicts of interest

Do any conflicts of interest arise?

YES ☐ **NO** ☒

If YES, full ethics review required
For example:
Might research objectivity be compromised by sponsorship?
Might any issues of intellectual property or roles in research be raised?

Funding

Is your research funded externally?

YES ☐ **NO** ☒

If YES, does the funder appear on the ‘currently automatically approved’ list on the UTREC website?

YES ☐ **NO** ☐

If NO, you will need to submit a Funding Approval Application as per instructions on the UTREC website.

Research with animals

Does your research involve the use of living animals?

YES ☐ **NO** ☒

If YES, your proposal must be referred to the University's Animal Welfare and Ethics Committee (AWEC)

University Teaching and Research Ethics Committee (UTREC) pages
http://www.st-andrews.ac.uk/utrec/

Bibliography

[1] Blood Cancer UK, "Blood cancer uk — facts and information about blood cancer," https://bloodcancer.org.uk/news/blood-cancer-facts/, 2019, aug. 18, 2019. Accessed: Oct. 25, 2024.

[2] "Cancer research uk," https://www.cancerresearchuk.org/health-professional/cancer, accessed: Oct. 25, 2024.

[3] The, "An expert-annotated dataset of bone marrow cytology in hematologic malignancies," https://www.cancerimagingarchive.net/collection/bone-marrow-cytomorphology_mll_helmholtz_fraunhofer/, 2024, the Cancer Imaging Archive (TCIA), Sep. 26, 2024. Accessed: Oct. 25, 2024.

[4] L. Guo *et al.*, "A method to classify bone marrow cells with rejected option," *Biomedical Engineering / Biomedizinische Technik*, vol. 67, no. 3, pp. 227–236, Apr 2022.

[5] C. Stryker and E. Kavlakoglu, "What is artificial intelligence (ai)?" IBM, 08 2024. [Online]. Available: https://www.ibm.com/think/topics/artificial-intelligence

[6] G. Currie, K. E. Hawk, E. Rohren, A. Vial, and R. Klein, "Machine learning and deep learning in medical imaging: Intelligent imaging," *Journal of Medical Imaging and Radiation Sciences*, vol. 50, 10 2019.

[7] M. P. McBee, O. A. Awan, A. T. Colucci, C. W. Ghobadi, N. Kadom, A. P. Kansagra, S. Tridandapani, and W. F. Auffermann, "Deep learning in radiology," *Academic Radiology*, vol. 25, no. 11, pp. 1472–1480, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1076633218301041

[8] C. P. Langlotz, B. Allen, B. J. Erickson, J. Kalpathy-Cramer, K. Bigelow, T. S. Cook, A. E. Flanders, M. P. Lungren, D. S. Mendelson, J. D. Rudie, G. Wang, and K. Kandarpa, "A roadmap for foundational research on artificial intelligence in medical imaging: From the 2018 nih/rsna/acr/the academy workshop," *Radiology*, vol. 291, pp. 781–791, 06 2019. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/30990384/

[9] T. T. D. Team *et al.*, "Theano: A python framework for fast computation of mathematical expressions," https://arxiv.org/abs/1605.02688, 2016, arXiv:1605.02688 [cs], May 2016. Accessed: Jul. 03, 2022.

[10] "Blood cancers," https://www.cancerresearchuk.org/about-cancer/blood-cancers/, 2024, mar. 04, 2024. Accessed: Oct. 25, 2024.

[11] "Blood cancer uk — blood cancer types," https://bloodcancer.org.uk/understanding-blood-cancer/blood-cancer-types/, accessed: Oct. 25, 2024.

[12] "Blood cancers: Symptoms, diagnosis and treatment," https://www.yalemedicine.org/conditions/blood-cancers, yale Medicine. Accessed: Oct. 25, 2024.

[13] S. Lee, "The plasma cells," https://cancer.ca/en/cancer-information/cancer-types/multiple-myeloma/what-is-multiple-myeloma/the-plasma-cells, 2014, canadian Cancer Society, 2014. Accessed: Oct. 25, 2024.

[14] Leukemia & Lymphoma Society, "Facts and statistics — leukemia and lymphoma society," https://www.lls.org/facts-and-statistics/facts-and-statistics-overview, 2015, mar. 03, 2015. Accessed: Oct. 25, 2024.

[15] "Blood cancers," https://www.bms.com/assets/bms/us/en-us/pdf/Disease-State-Info/blood-cancers-at-a-glance.pdf, accessed: Oct. 25, 2024.

[16] J. Huang *et al.*, "Disease burden, risk factors, and trends of leukaemia: A global analysis," *Frontiers in Oncology*, vol. 12, no. 904292, Jul 2022.

[17] U. F. O. Themes, "Normal bone marrow cells: Development and cytology," https://basicmedicalkey.com/normal-bone-marrow-cells-development-and-cytology/, 2017, ba-

sicmedical Key, Feb. 19, 2017. Accessed: Oct. 25, 2024.

[18] M. Markman, "What is cancer pathology?" https://www.cancercenter.com/cancer-types/leukemia/diagnosis-and-detection, 2019, cancer Treatment Centers of America, Oct. 22, 2019. Accessed: Oct. 25, 2024.

[19] M. Mandal, "Cnn for deep learning — convolutional neural networks (cnn)," https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/, 2021, analytics Vidhya, May 01, 2021. Accessed: Oct. 25, 2024.

[20] "Image classification using machine learning," https://www.analyticsvidhya.com/blog/2022/01/image-classification-using-machine-learning/, 2022, analytics Vidhya, Jan. 20, 2022. Accessed: Oct. 25, 2024.

[21] D. Bhardwaj, "Why cnn performs better than ann on image classification," https://medium.com/@divyanshub2311/why-cnn-performs-better-than-ann-on-image-classification-7f92e5a92904, 2022, medium, Dec. 17, 2022. Accessed: Oct. 25, 2024.

[22] "Cnn image classification — image classification using cnn," https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets/, 2020, analytics Vidhya, Feb. 18, 2020. Accessed: Oct. 25, 2024.

[23] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *Journal of Big Data*, vol. 6, no. 1, Dec 2019.

[24] S. Takahashi *et al.*, "Comparison of vision transformers and convolutional neural networks in medical image analysis: A systematic review," vol. 48, no. 1, Sep 2024.

[25] "Convolutional neural networks (cnn) and deep learning," https://www.intel.com/content/www/us/en/internet-of-things/computer-vision/convolutional-neural-networks.html, intel, 2020.

[26] B. Artley, "Mnist: Keras simple cnn (99.6%) - brendan artley - medium," https://medium.com/@BrendanArtley/mnist-keras-simple-cnn-99-6-731b624aee7f, 2022, medium, Apr. 27, 2022. Accessed: Oct. 25, 2024.

[27] IBM, "What are convolutional neural networks? — ibm," https://www.ibm.com/topics/convolutional-neural-networks, 2024, accessed: Oct. 25, 2024.

[28] D. Sharma, "Cnn for image classification — image classification using cnn," https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/, 2021, analytics Vidhya, Jan. 11, 2021. Accessed: Oct. 25, 2024.

[29] M. Mishra, "Convolutional neural networks, explained," https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939, 2020, medium, Aug. 27, 2020. Accessed: Oct. 25, 2024.

[30] S. Saha, "A comprehensive guide to convolutional neural networks — the eli5 way," https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53, 2018, medium, Dec. 15, 2018. Accessed: Oct. 25, 2024.

[31] K. Kalra, "Convolutional neural networks for image classification," https://medium.com/@khwabkalra1/convolutional-neural-networks-for-image-classification-f0754f7b94aa, 2023, medium, Jul. 14, 2023. Accessed: Oct. 25, 2024.

[32] H. Yu, L. T. Yang, Q. Zhang, D. Armstrong, and M. J. Deen, "Convolutional neural networks for medical image analysis: State-of-the-art, comparisons, improvement and perspectives," *Neurocomputing*, vol. 444, pp. 92–110, Jul 2021.

[33] H. Jia, J. Zhang, K. Ma, X. Qiao, L. Ren, and X. Shi, "Application of convolutional neural networks in medical images: a bibliometric analysis," *Quantitative Imaging in Medicine and Surgery*, vol. 14, no. 5, pp. 3501–3518, May 2024.

[34] C. Matek, S. Krappe, C. Münzenmayer, T. Haferlach, and C. Marr, "Highly accurate differentiation of bone marrow cell morphologies using deep neural networks on a large

image data set," *Blood*, vol. 138, no. 20, pp. 1917–1927, Nov 2021.

[35] A. Erdoğan, "Resnext: A new paradigm in image processing," https://medium.com/@atakanerdogan305/resnext-a-new-paradigm-in-image-processing-ee40425aea1f, 2023, medium, Nov. 04, 2023. Accessed: Oct. 25, 2024.

[36] R. M. Tayebi *et al.*, "Automated bone marrow cytology using deep learning to generate a histogram of cell types," *Communications Medicine*, vol. 2, no. 1, Apr 2022.

[37] A. S. Abeed, A. Atiq, A. A. Anjum, A. A. Efat, and D. Z. Karim, "Bomacnet: A convolutional neural network model to detect bone marrow cell cytology," in *Proceedings of the IEEE Conference*, 2022, doi:10.1109/iccit57492.2022.10054976.

[38] "Dense layers," https://analyticsindiamag.com/topics/what-is-dense-layer-in-neural-network/#h-what-is-a-dense-layer, 2024, aIM, Aug. 2024. Accessed: Oct. 28, 2024.

[39] "Average pooling - an overview — sciencedirect topics," https://www.sciencedirect.com/topics/computer-science/average-pooling, accessed: Oct. 28, 2024.

[40] B. Ananthakrishnan, A. Shaik, S. Akhouri, P. Garg, V. Gadag, and M. S. Kavitha, "Automated bone marrow cell classification for haematological disease diagnosis using siamese neural network," *Diagnostics*, vol. 13, no. 1, p. 112, Dec 2022.

[41] K. H. Prodhan, I. A. Amin, A. J. Das, and M. M. Uddin, "Bone marrow classification using hematologic malignancies dataset," in *2023 26th International Conference on Computer and Information Technology (ICCIT)*, vol. 27, Dec 2023, pp. 1–6.

[42] H. He *et al.*, "A novel bone marrow cell recognition method based on multi-scale information and reject option," *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108540, Jul 2024.

[43] "Imagenet," www.image-net.org. [Online]. Available: https://www.image-net.org

[44] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, no. 4, pp. 611–

629, Jun 2018.

[45] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, no. 1, pp. 60–88, Dec 2017.

[46] S. Bao, P. Wang, and Albert, "3d randomized connection network with graph-based inference," in *Lecture Notes in Computer Science*, Jan 2017, pp. 47–55.

[47] A. Esteva *et al.*, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, Jan 2017.

[48] M. I. Razzak, S. Naz, and A. Zaib, "Deep learning for medical image processing: Overview, challenges and future," https://doi.org/10.48550/arxiv.1704.06825, 2017, arXiv, Apr. 2017.

[49] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Deeplearningbook.org, 2016, accessed: Oct. 30, 2024. [Online]. Available: https://www.deeplearningbook.org/

[50] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Systems with Applications*, vol. 73, pp. 220–239, May 2017.

[51] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in nlp," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[52] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," https://arxiv.org/abs/1702.08608, 2017, arXiv:1702.08608 [cs, stat], Mar. 2017.

[53] A. Barredo Arrieta *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information Fusion*, vol. 58, no. 1, pp. 82–115, Jun 2020.

[54] S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane, "Adversarial attacks on medical machine learning," *Science*, vol. 363, no. 6433, pp. 1287–1289, Mar 2019.

[55] A. Vaswani *et al.*, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008. [Online]. Available: https://papers.nips.cc/paper/7181-attention-is-all-you-need

[56] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," https://openreview.net/pdf?id=YicbFdNTTy, accessed: date not provided.

[57] "Google scholar," https://scholar.google.com/scholar_lookup?title=An%20image%20is%20worth%2016x16%20words%3A%20transformers%20for%20image%20recognition%20at%20scale&publication_year=2021&author=A.%20Dosovitskiy&author=L.%20Beyer&author=A.%20Kolesnikov, 2020, 2020. Accessed: Oct. 25, 2024.

[58] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision – ECCV 2020*, vol. 12346, 2020, pp. 213–229.

[59] J. Chen *et al.*, "Transunet: Transformers make strong encoders for medical image segmentation," https://arxiv.org/abs/2102.04306, 2021, arXiv:2102.04306 [cs], Feb. 2021.

[60] W. Wang *et al.*, "Crossformer: A versatile vision transformer hinging on cross-scale attention," https://arxiv.org/abs/2108.00154, 2021, arXiv preprint. Accessed: Oct. 25, 2024.

[61] S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane, "Adversarial attacks on medical machine learning," *Science*, vol. 363, no. 6433, pp. 1287–1289, Mar 2019.

[62] V. Bastos and W. R. Schwartz, "Bubblenet: A disperse recurrent structure to recognize activities," in *Proceedings of the IEEE International Conference on Image Processing*, vol. abs 1904 11451, Sep 2020, pp. 2216–2220.

[63] L. Gao, L. Zhang, C. Liu, and S. Wu, "Handling imbalanced medical image data: A deep-learning-based one-class classification approach," *Artificial Intelligence in Medicine*, vol. 108, p. 101935, Aug 2020.

[64] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, "A survey on addressing high-class imbalance in big data," *Journal of Big Data*, vol. 5, no. 1, Nov 2018.

[65] D. B. Or, "Solving the class imbalance problem," https://medium.com/metaor-artificial-intelligence/solving-the-class-imbalance-problem-58cb926b5a0f, 2024, metaOr Artificial Intelligence, Jan. 27, 2024. Accessed: Oct. 29, 2024.

[66] M. Xu, S. Yoon, A. Fuentes, and D. S. Park, "A comprehensive survey of image augmentation techniques for deep learning," *Pattern Recognition*, vol. 137, p. 109347, May 2023.

[67] F. Liu, "Evaluating the impact of data augmentation on explainable ai in medical image analysis," https://pure.tue.nl/ws/portalfiles/portal/333611106/Liu_X_2_.pdf, 2024, accessed: Oct. 25, 2024.

[68] T. Zuo *et al.*, "Automated classification of papillary renal cell carcinoma and chromophobe renal cell carcinoma based on a small computed tomography imaging dataset using deep learning," *Frontiers in Oncology*, vol. 11, Nov 2021.

[69] J. Tang, M. Sharma, and R. Zhang, "Explaining the effect of data augmentation on image classification tasks," https://cs231n.stanford.edu/reports/2022/pdfs/57.pdf, 2022, accessed: Oct. 25, 2024.

[70] IBM, "What is overfitting? — ibm," https://www.ibm.com/topics/overfitting, 2024, accessed: Oct. 25, 2024.

[71] P. Hallaj, "Data augmentation: Benefits and disadvantages," https://medium.com/@pouyahallaj/data-augmentation-benefits-and-disadvantages-38d8201aead, 2023, medium, Sep. 20, 2023. Accessed: Oct. 25, 2024.

[72] K. Faryna, van, and G. Litjens, "Automatic data augmentation to improve generalization of deep learning in h&e stained histopathology," *Computers in Biology and Medicine*, vol. 170, p. 108018, Mar 2024.

[73] X. Wang, V. Liesaputra, Z. Liu, Y. Wang, and Z. Huang, "An in-depth survey on deep learning-based motor imagery electroencephalogram (eeg) classification," *Artificial Intelligence in Medicine*, vol. 147, p. 102738, Jan 2024.

[74] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol. 97, p. 105524, May 2019.

[75] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1409.1556

[76] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1512.03385

[77] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," 2018. [Online]. Available: https://arxiv.org/abs/1608.06993

[78] W. Huang, X. Hu, S. Abousamra, P. Prasanna, and C. Chen, "Hard negative sample mining for whole slide image classification," 2024. [Online]. Available: https://arxiv.org/abs/2410.02212

[79] IBM, "Overfitting," Ibm.com, 10 2021. [Online]. Available: https://www.ibm.com/think/topics/overfitting

[80] M. Gu, Y. Zhang, Y. Wen, G. Ai, H. Zhang, P. Wang, and G. Wang, "A lightweight convolutional neural network hardware implementation for wearable heart rate anomaly detection," *Computers in Biology and Medicine*, vol. 155, p. 106623, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482523000884

[81] "sklearn.metrics," https://scikit-learn.org/stable/api/sklearn.metrics.html, n.d., scikit-learn.

[82] M. Hossin and S. M.N, "A review on evaluation metrics for data classification evaluations," *International Journal of Data Mining Knowledge Management Process*, vol. 5, pp. 01–11, 03 2015.

[83] M. Grandini, E. Bagli, and G. Visani, "Metrics for multi-class classification: an overview," 2020. [Online]. Available: https://arxiv.org/abs/2008.05756

[84] R. Egele, F. Mohr, T. Viering, and P. Balaprakash, "The unreasonable effectiveness of early discarding after one epoch in neural network hyperparameter optimization,"

Neurocomputing, vol. 597, p. 127964, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231224007355

[85] T. Gonsalves and J. Upadhyay, "Chapter eight - integrated deep learning for self-driving robotic cars," in *Artificial Intelligence for Future Generation Robotics*, R. N. Shaw, A. Ghosh, V. E. Balas, and M. Bianchini, Eds. Elsevier, 2021, pp. 93–118. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780323854986000101

[86] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017. [Online]. Available: https://arxiv.org/abs/1412.6980

[87] D. Kingma and J. Lei Ba, "Adam: A method for stochastic optimization," 2015. [Online]. Available: https://arxiv.org/pdf/1412.6980

[88] R. Zaheer and H. Shaziya, "A study of the optimization algorithms in deep learning," in *2019 Third International Conference on Inventive Systems and Control (ICISC)*, 2019, pp. 536–539.