

A microscopic image of malaria parasites (erythrocytes) with numerous bounding boxes overlaid. Some bounding boxes are red, indicating detected parasites, while others are grey, indicating non-detected or false detections. The background is a light purple/pinkish hue.

Automating malaria diagnosis: a machine learning approach

Noor van Driel

Master of Science Thesis

Automating malaria diagnosis: a machine learning approach

**Erythrocyte segmentation and parasite identification in thin blood
smear microscopy images using convolutional neural networks**

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Systems and Control at Delft
University of Technology

Noor van Driel

December 8, 2020

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of
Technology



Copyright © Delft Center for Systems and Control (DCSC)
All rights reserved.



Abstract

Reliable malaria diagnosis techniques that are suitable for point-of-care testing in high burden areas, are vital for effective treatment and monitoring of the disease. Identification of malaria parasites in Giemsa stained blood slides is currently the most widely accepted technique, but its availability is limited by the need for highly trained experts to interpret the data.

In this work, a two stage automated image classification strategy is proposed, to eliminate this dependency on human expertise. Blood slides that were photographed at $20\times$ magnification were used in our experiments, allowing for a larger Field of View than regular thin film microscopy at $100\times$. Erythrocytes are first localised and segmented by a Convolutional Neural Network, the architecture of which is based on U-Net, with some adaptations and improvements made for our purposes. The sensitivity and positive predictive value of the localisation were both 0.998, resulting in accurate cell counts. A transfer learning strategy, in which the existing VGG-16 network is used as a feature extractor and combined with a new fully connected layer to predict correct activations for our classification, is then used to classify the segmented erythrocytes as either infected with *Plasmodium Falciparum* parasite or healthy. Sensitivity and specificity of the predicted classification were 0.795 and 0.915 respectively. It is concluded that, although this method may not fully eliminate the need for trained experts, the algorithms proposed can be of great assistance in aiding the diagnostic decision making process.

Table of Contents

Acknowledgements	xi
1 Introduction	1
1-1 Available diagnostic methods and their limitations	2
1-1-1 Light microscopy with stained blood smears	2
1-1-2 Fluorescence microscopy	4
1-1-3 Rapid Diagnostic Tests	5
1-1-4 Polymerase Chain Reaction	5
1-1-5 Experimental malaria tests	6
1-2 Research objective	7
2 Background	9
2-1 Conventional malaria image analysis techniques	9
2-2 Neural Networks and malaria image analysis	12
2-2-1 Mathematical principles	13
2-2-2 Neural networks applied to malaria image data	18
2-3 Discussion of automated malaria diagnosis techniques	19
3 Method	21
3-1 Datasets	21
3-2 Threshold based segmentation	23
3-3 U-Net based segmentation	27
3-3-1 Architecture	27
3-3-2 Training data	29
3-3-3 Training and testing strategy	30
3-4 Classification	32
3-4-1 Network architecture	32
3-4-2 Training and testing strategy	32

4 Results	35
4-1 Segmentation results	35
4-1-1 Segmentation results with threshold method	35
4-1-2 Segmentation results with U-Net method	39
4-2 Classification results	42
4-2-1 Results for network trained on Rajaraman data	42
4-2-2 Results for network trained on AiDx data	46
5 Discussion	49
6 Conclusion	53
Bibliography	55
Glossary	63
List of Acronyms	63

List of Figures

1-1	Map of malaria cases in 2018 (per 1000 population at risk), from [1]	1
1-2	Giemsa stained blood smears of peripheral blood infected with <i>P. falciparum</i> parasites, as seen under 100x oil immersion objective. [2]	3
2-1	Schematic representation of the basic image analysis pipeline followed by most (traditional) automated malaria diagnosis algorithms, the numbers underneath the arrows refer to the four operations in this pipeline; 1) preprocessing, 2) segmentation, 3) feature extraction and 4) classification.	10
2-2	Schematic depiction of a feed forward neural network with three inputs, two outputs and one hidden layer. On the right side, the general architecture of a single neuron is depicted.	13
2-3	Different activation functions used by artificial neurons.	14
2-4	Schematic depiction of the convolution of a 6×6 input image with a 3×3 kernel. In order to produce a 6×6 feature map, padding is used.	15
2-5	Upper image: convolution of a 3×3 input matrix with a 2×2 kernel to create a 2×2 feature map, expressed as a matrix operation. Lower image: transposed convolution of a 2×2 input image with that kernel, to create a 4×4 feature map, expressed as a matrix operation.	16
3-1	Four images from the Loddo dataset of Giemsa stained thin blood smears infected with <i>P. Falciparum</i> , taken with $100 \times$ oil immersed objective.	22
3-2	Six images from each of the two classes in the Rajaraman dataset	23
3-3	Two images from the AidX dataset of Giemsa stained thin blood smears infected with <i>P. Falciparum</i> , taken with $20 \times$ objective.	23
3-4	Schematic depiction of the segmentation algorithm, which pre-processes the blood film images (step 1-2), divides them into binary classes (step 3), separates the resultant objects through watershedding (step 4) and crops them out of the original image (step 4-5).	24
3-5	Visualisation of the watershedding algorithm; a) shows the original thin blood film with several overlapping cells, b) shows the binary mask based on this image, c) shows the euclidean distance map of this binary image, with the ultimate eroded points, and d) shows the resultant binary mask after watershedding.	26

3-6	Full network architecture used for creating a segmentation map for a 256×256 RGB input image. The arrows denote convolutional (conv) and sampling operations, and the blocks denote the output of each operation. The size (width \times height) is written next to the levels, and the number of channels or depth is written above each block.	28
3-7	Left: 256×256 tile cropped out of an image in the AiDx dataset. Right: Hand-drawn binary mask used for training.	30
3-8	Three different augmented training samples and their corresponding masks, created by applying three sets of transformations to the image and mask in figure 3-7.	31
3-9	The architecture used for classification of segmented erythrocytes. The arrows denote convolutional (conv), pooling and fully connected (FC) operations, and the blocks denote the output of each operation. The output size is written above the blocks (width \times height \times depth), note that the output of the fully connected layers is vectorised so one-dimensional. The convolutional layers of the VGG-16 net and their pre-trained weights on the ImageNet database are not adjusted, only the final two fully connected layers are.	33
3-10	Two parasitised erythrocytes from the AiDx images resized to 100×100 , with three augmented training samples based on on each. Original, non-augmented images are shown on the left.	34
4-1	Segmentation result for image no. 1 in the Loddo dataset. Three wrong results are pointed out; at 1) over-segmentation occurred, resulting in multiple small objects, which were all removed before applying the segmentation mask, resulting in a false negative (FN), at 2) a non-erythrocyte object was found (false positive (FP)) and at 3), under-segmentation occurred (FN).	36
4-2	Error of threshold segmentation algorithm in image no. 5 from the AiDx dataset; a) is a section of the original image, with ground truth points overlaid, b) is the corresponding binary segmentation mask and c) shows the result of applying the segmentation mask to the original image, with hits and misses.	39
4-3	Progression of loss and accuracy of the U-Net during training. Both performance metrics were calculated at the end of each iteration on the batch of augmented training data that was used at that step. The values in this graph are the average over the three batches in the epoch.	39
4-4	Image no.5 of the AiDx dataset, with segmentation overlay produced by U-Net algorithm. To allow for comparison between the two segmentation methods, the square indicates the area for which the results of the threshold algorithm were depicted in figure 4-2.	40
4-5	Progression of loss and accuracy during training of the classification network on the Rajaraman data. The loss is plotted on a logarithmic scale.	42
4-6	Confusion matrices for both validation sets, showing the total number of correct and incorrect predictions made on each set.	43
4-7	Receiver Operating Characteristic curves for both validation sets. Sensitivity and specificity are plotted for different thresholds, starting at $\tau = 1$ on the left (all objects are assigned to class uninfected) and ending at $\tau = 0$ on the right (all objects are assigned to class parasitised) Area under the curve is given in lower right corner. The reference line depicts a theoretical 'random guessing' classifier (i.e. on that is right 50 % of the time at $\tau = 0.5$).	44
4-8	Predicted locations of healthy erythrocytes, indicated with a gray bounding box, and parasitised erythrocytes, indicated by a red bounding box, in image no. 6 from the AiDx dataset.	45
4-9	Progression of loss and accuracy during training of the classification network on the AiDx data.	46

-
- 4-10 Confusion matrices showing the number of correct and incorrect predictions on erythrocytes from AiDx image no. 6, before and after retraining the network. . . . 47
- 4-11 Locations of cells as estimated by the U-Net algorithm, combined with locations of parasites as estimated by the retrained VGG-16 based classifier, in image no. 6 from the AiDx dataset. Predicted parasite locations are indicated by a red bounding box; TP cells are indicated with a solid line while FP cells are indicated with a dotted line. FN cells are indicated by a blue bounding box and TN cells indicated with a gray bounding box. To compare performance, see 4-8, in which the prediction by the network on this same image before retraining is depicted. 48

List of Tables

1-1	Thin smear images of each of the four life stages that appear in human erythrocytes infected with different species of Plasmodium parasites. Based on [3, 4], images from [2].	4
2-1	Summary of automated malaria classification methods. Performance is given in terms of accuracy (acc) or sensitivity (sens) and specificity (spec), depending on what was reported.	20
2-2	Performance requirements for World Health Organization (WHO) microscopist competence levels, from [5].	20
3-1	Table of tunable parameters in the algorithm, and the values that were used for segmenting both datasets.	27
4-1	Test results for threshold based segmentation algorithm on the Loddo dataset. For each individual image, the number of objects that were true positive (TP), FP and FN are given, and the performance measures were calculated from those.	38
4-2	Test results for threshold based segmentation algorithm on the AidX dataset. For each individual image, the number of objects that were TP, FP and FN are given, and the performance measures were calculated from those.	38
4-3	Test results AiDx dataset U-Net based segmentation	40
4-4	Performance measures of the trained network on the Rajaraman validation set and the segmented erythrocytes from the AiDx dataset.	43
4-5	Ground truth cell and infection counts, compared with estimates for both by the U-Net based segmentation method and the VGG-16 based classifier respectively. Ground truth and estimated parasiteamia levels are given, and compared in the final column.	45
4-6	Performance measures on erythrocytes from AiDx image no. 6, before and after retraining the network.	47
4-7	Ground truth parasiteamia for AiDx image no. 6, compared with prediction by network before and after retraining	47

Acknowledgements

I would like to thank my supervisors prof.dr. G. Vdovin and dr.ir. T.E. Agbana for helping me throughout my research project. I enjoyed the discussions we've had, and I think the feedback you gave me was always sharp and to the point.

Most of all, I think the research you do inspires and has the potential to make real societal impact, and I am grateful to have had the opportunity to be a part of it.

I would also like to thank everyone else working at AiDx, for providing me with the image data that formed the basis for my experiments.

Delft, University of Technology
December 8, 2020

Noor van Driel

Chapter 1

Introduction

Malaria is a serious, but curable disease that affects millions of people every year. It is caused by infection with a parasite of the Plasmodium genus, five species of which (*P. falciparum*, *P. vivax*, *P. ovale*, *P. malariae* and *P. knowlesi*) can infect humans. Primarily, these parasites are transmitted to humans via the bite of female Anopheles mosquitoes, which act as the disease vector. After entering a person's blood, they develop and multiply, first in the liver cells and then in the erythrocytes. The erythrocytic parasites are responsible for the clinical manifestations of the disease. Common symptoms include fever, fatigue, nausea and headaches. However, in severe cases, the infection can result in organ failure and death. Most severe cases occur after infection with the falciparum parasite [6].

In 2018 alone, the World Health Organization (WHO) reported an estimated 228 million cases and 405 000 deaths. Figure 1-1 shows the global case incidence rate of malaria. The burden of the disease concentrates heavily in sub-Saharan African countries and India, together they account for 85% of fatal cases. Two-thirds of total fatalities were children under five [1].

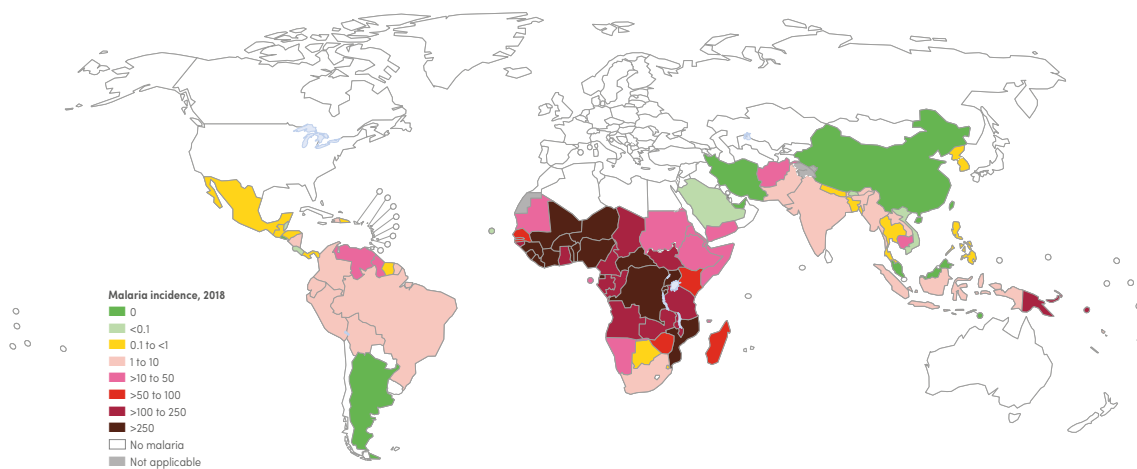


Figure 1-1: Map of malaria cases in 2018 (per 1000 population at risk), from [1]

When the infection is diagnosed early, it can be treated effectively with medication, preventing a mild case of malaria from developing into a life-threatening one. Accurate malaria diagnosis is crucial for proper treatment and disease monitoring. However, in the areas where the malaria burden is the greatest, access to diagnostic methods is most limited, due to limited resources and remoteness [7]. Clinical diagnosis of malaria (based on symptoms, such as the presence of a fever, rather than a diagnostic test), requires the least amount of resources and is therefore still widely practised. However, malaria symptoms are varied and overlap with many other common tropical diseases, which leads to low specificity in clinical diagnosis. False positives are common in highly endemic regions, which not only leads to the real cause of the symptoms going untreated, but also to over-prescription of antimalarial drugs, which introduces unnecessary side effects and contributes to drug resistance in parasites [8, 9, 10].

The WHO therefore recommends that all cases of suspected malaria should be confirmed with a parasitological test. The number of cases for which this happens has dramatically increased in sub-Saharan African countries over the past decade, from 38% of all suspected cases in 2010, to 85 % in 2018, mainly due to the development of Rapid Diagnostic Tests (RDTs). However, this trend is not universal; in some countries, tests are still only administered in 50% of cases [1]. Furthermore, the increase in testing has come to a stop over recent years. It is therefore evident that the development of new, accurate diagnostic methods that can be used for point-of-care testing in resource-limited, on-field settings is still necessary.

We will now first give an overview of existing diagnostic methods. Their advantages and limitations are discussed, in order to identify relevant opportunities for improvement. From this, the research objective of this thesis follows, which will be introduced in section 1-2.

1-1 Available diagnostic methods and their limitations

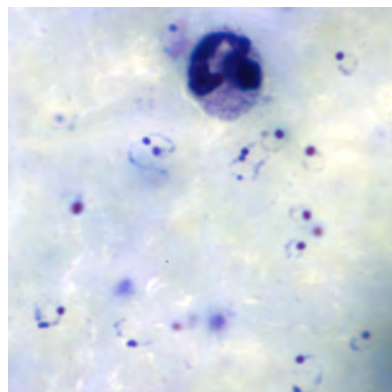
1-1-1 Light microscopy with stained blood smears

Parasite identification through light microscopy inspection of blood smears is currently the recommended method for diagnosing malaria accurately. The standard microscopic diagnostic procedure, as recommended by the WHO, consists of the following steps [5]:

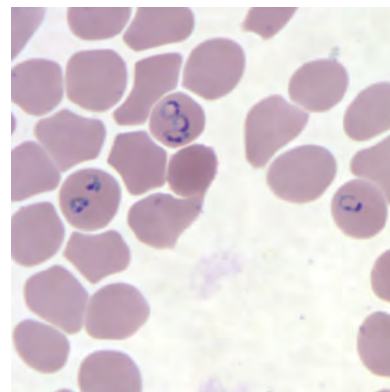
1. **Preparation of blood films:** A sample of the patient's peripheral blood is acquired, usually from the finger. The blood is applied to a microscopic slide, and a thick- or thin film is prepared. Thick films consist of multiple layers of blood cells, while thin films are spread out such that only one layer of erythrocytes is present.
2. **Staining:** To allow for distinction of the blood cells and parasites, a stain is applied. Microscopy slides are most commonly stained with a Giemsa stain, but other Romanowsky stains such as Field's and Leishman stains can also be used. Field's stain has the advantage of very short staining time, but a slightly lower sensitivity is achieved when this stain is used [11]. Leishman staining takes half as long as Giemsa staining, and leads to the same diagnostic accuracy, but the solution is less stable [12, 13].
3. **Examination with light microscope:** The microscopic slides are examined with a 100x oil immersion objective. For thick films, at least 100 Field of View (FOV) should

be examined before a negative diagnosis is reached, while for thin films, the minimum is 800 FOV.

4. **Data interpretation:** In thick smears the parasite density is determined in parasites per μL of blood, by determining the number of parasites $\times 8000$, divided by the number of white blood cells. In thin smears the number of infected and non-infected red blood cells is tallied, and parasitaemia is expressed as percentage of total cells infected. The species and stage of the parasites is also identified.



(a) Thick smear with several ring stage parasites and one white blood cell.



(b) Thin smear with a single layer of red blood cells, three of which infected with multiple ring stage parasites.

Figure 1-2: Giemsa stained blood smears of peripheral blood infected with *P. falciparum* parasites, as seen under 100x oil immersion objective. [2]

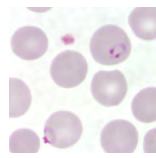
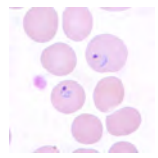
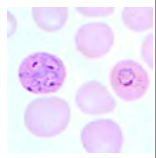

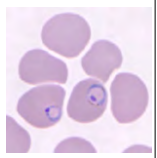
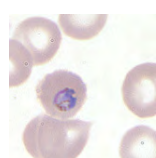
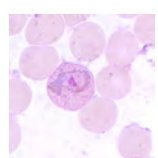
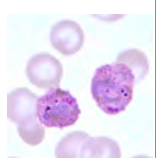
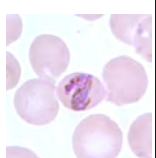
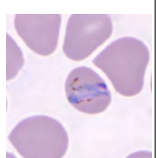

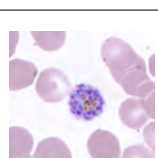
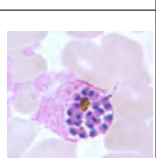
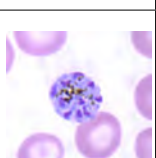
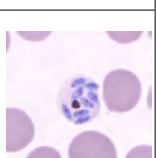

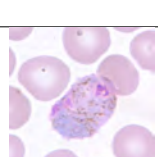
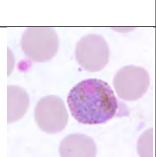
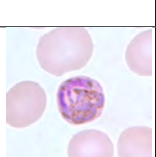
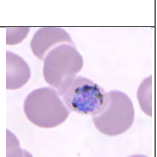
In figure 1-2 a thick- and thin smear of malaria infected blood is shown. The advantage of thick smears is that a larger amount of blood can be inspected in one FOV, so a high sensitivity can be reached in less time. However, when parasite density is high, it can be difficult to distinguish individual parasites, so thin smears are used. Accurate knowledge on parasitaemia is important for decisions on treatment administration. Hyperparasitaemia, which is defined as parasitaemia $> 4\%$, or $200\,000/\mu L$ in endemic regions ($> 2\%$ for non-immune people, such as travellers), is associated with severe malaria. Patients with hyperparasitaemia need to be closely monitored, if feasible in hospital, even when not suffering from severe symptoms, because risk of treatment failure is high [9].

Thin smears are also useful for reaching accurate conclusions on parasite species and life stage. Table 1-1 summarises the differences in appearance between all *Plasmodium* parasites that can infect humans, in their four erythrocytic life stages.

The sensitivity of light microscopy diagnosis is limited by the number of FOV examined. When the WHO recommended procedure is followed, the theoretical detection limit is 5 parasites $/\mu l$ of blood, making for a very sensitive test. This, along with the possibility to obtain species and life stage information, are the main advantages of this diagnostic method. Furthermore, the relatively inexpensive equipment makes this the most cost-effective test currently available for highly endemic settings, even when not taking into consideration that light microscopy equipment can also be used for the diagnosis of other diseases [14].

However, the preparation of the stained blood slides and the data interpretation are labour intensive and require highly trained experts [8, 15]. In practice, the theoretical detection limit

Table 1-1: Thin smear images of each of the four life stages that appear in human erythrocytes infected with different species of Plasmodium parasites. Based on [3, 4], images from [2].

	P. Falciparum	P. Vivax	P. Ovale	P. Malariae	P. Knowlesi	
Ring stage	Characterised by a cytoplasm ring with one or two chromatin dots. P. falciparum and P. knowlesi may appear on periphery of cell (appliqué/allocé form). Cells infected with P. vivax and P. ovale can be slightly enlarged. "Bird's-eye" form is characteristic for P. malariae.					
Trophozoite	Parasites have matured and the pigmented cytoplasm has grown. For P. falciparum, they appear as larger, thicker circles. Cells with P. vivax and P. ovale trophozoites are often enlarged and appear speckled (Schüffner's dots). Band forms are characteristic for P. malariae and P. knowlesi.					
Schizont	Contain anywhere between 6 and 24 merozoites, which appear as chromatin dots clustered around dark-brown pigment. Larger amounts (> 12) are typical for P. vivax. P. falciparum merozoites are slightly smaller and don't fill the cell completely. P. malariae pigment is coarse.					
Gametocyte	Gametocytes of most species are round to oval and fill the host cell. For P. falciparum, however, they have a crescent shape, which distorts and enlarges the host cell.					

is rarely reached. A non-expert microscopist can commit many errors when interpreting the data; misidentification of the parasite species is common, as well as seriously underestimating the parasite count [16]. Best estimates for the sensitivity and specificity of microscopic diagnosis in district-hospitals and health-centre general labs in sub-Saharan countries are only 82% and 85% respectively [14].

The applicability of this method in on-field settings is further limited by the required maintenance, the cost and the need for electricity of the microscopy equipment. To address this limitation, several simple, portable microscopes which are battery operated have been developed over the years [17]. For example, Agbana et. al proposed that the wide-spread availability and advances in imaging capabilities of mobile phones could be leveraged on, by attaching an oil immersed ball lens onto the built-in camera, which effectively turns a mobile phone into a microscope [18].

1-1-2 Fluorescence microscopy

As an alternative to light microscopy, several diagnostic methods using fluorescence microscopy have been proposed. In these methods, a fluorochrome that attaches to the nucleic acids of the malaria parasite is used to stain a blood sample [19, 20]. The blood sample is then examined under a fluorescence microscope, which only emits light at the excitation wavelength of the fluorochrome. The advantage of using fluorescence microscopy over standard light microscopy is that a microscopist can easily spot the light-emitting parasites, and that

the interpretation can thus be done more quickly [21].

Of all fluorescence microscopy methods that have been published, the Quantitative Buffy Coat (QBC) method, in which blood samples are placed in acridine orange stained capillary tubes and centrifuged before examination, has most widely gained acceptance. The theoretical detection limit of this technique is about the same as that of light microscopy, but in several field experiments it was shown for low parasitaemia to outperform light microscopy in terms of sensitivity [19, 22, 23]. However, since acridine orange (AO) is a non-specific stain, meaning other cell nuclei also fluoresce when they are dyed with this stain, difficulties can arise in distinguishing the parasites, which limits the sensitivity. When the patient is infected with *P. falciparum*, this test offers excellent specificity (> 93%), but for other species, specificity was found to be very low compared with light microscopy (52%) [24]. Furthermore, this method is technically demanding, requiring specialised equipment to separate the cell layers by centrifugation, and it is not possible to accurately determine parasite species and parasitaemia with this test.

1-1-3 Rapid Diagnostic Tests

In order to reduce diagnostic complexity and allow for point-of-care testing, RDTs were developed. RDTs are lateral flow immunochromatographic tests, that detect the antigens produced in human blood by the presence of malaria parasites. They consist of a strip of nitrocellulose, with some dye-labelled antibody specific for the target antigen on one end, and a test line of antibody on the other end. A sample of the patient's blood is collected by a prick to finger, mixed with a buffer and applied to the dye-labelled antibody end. If malaria antigens are present in the blood, these labelled antibodies will attach to them, be carried over to the strip by the buffer and accumulate on the test line. This produces a visible line, which indicates a positive test result. [24]

Several WHO-qualified RDTs are available commercially and have been tested extensively in laboratory and field. The most commonly used ones can only detect *P. falciparum* or *P. vivax*, but tests that can detect multiple species and even distinguish between them are also available. A very crude estimation of the parasitaemia can be made by the intensity of the test line. The main advantage of RDTs is that no expertise is required to administer and interpret them, which means they can even be used for self-diagnosis [25]. Furthermore, they only take 15-30 minutes to process and require no electricity or additional equipment, making them uniquely suitable for in-field use.

However, RDTs have several limitations. They are less sensitive (detection limit ± 100 parasites / μl), making them unsuitable for detecting early-stage infections. Some commonly used target antigens remain present in the blood beyond the clearance of the parasites, severely limiting the specificity of these tests (51 %) and making them unsuitable for disease monitoring or detecting repeated infections [26]. The average cost per test in endemic settings, though low compared with other methods discussed in this section, is higher than that of light microscopy [14].

1-1-4 Polymerase Chain Reaction

Polymerase Chain Reaction (PCR) is a technique in which a specific strand of DNA is multiplied rapidly. In PCR-based malaria diagnostic tests, the blood of a patient is tested for the

presence of the DNA of a plasmodium species, by using a string of the parasitic RNA as the primer. After the multiplication process, the PCR products can be analysed by electrophoresis and a diagnosis is reached [27].

This diagnostic method has been shown to be more accurate than others, both in lab conditions and in field. It has excellent sensitivity, with a practical detection limit of 5 parasites/ μl of blood, and is highly specific; false positives can be ruled out when two PCR sets are used. It can also be used to accurately detect mixed infections and drug-resistant parasites. [10, 28].

The usefulness of this technique, particularly in settings where access to laboratory facilities is limited, is however limited by the complexity of the methodology, the need for highly specialised equipment and trained experts, and the time lag between sample collection and the diagnosis. In many of the promising field tests described in literature, the PCR was performed weeks after the blood sample acquisition, which obviously makes the results unusable in routine clinical practice [29].

As an alternative, loop-mediated isothermal amplification (LAMP) has been developed, which is claimed to combine the low costs and technical requirements of other diagnostic techniques with the sensitivity and specificity of PCR. [30, 31] However, clinical trials in field settings have thus far been limited [8, 32].

1-1-5 Experimental malaria tests

In literature, many additional novel diagnostic tests have been proposed. A multitude of alternative microscopy techniques, such as multispectral microscopy, quantitative phase imaging and Raman spectroscopy have been applied towards the identification of malaria parasites in experiments, however, none of these methods have gained clinical acceptance yet [33, 34].

Several tests aimed at detecting the presence of hemozoin, a disposal product formed by the plasmodium parasite in the blood, were proposed. Examples include the use of flow cytometry and laser desorption mass spectrophotometry [35, 36]. Both methods show promise in terms of sensitivity and specificity, but require highly specialised diagnostic equipment and personnel.

Recently, as an alternative to PCR and LAMP, some novel methods to detect specific parasitic DNA sequences were proposed. One example of these are methods which makes use of a surface-enhanced Raman scattering (SERS) platform. These methods are claimed to have excellent sensitivity and specificity, and be suitable for integration into portable platforms, however, no field tests have been published yet [37, 38].

In addition to diagnostic tests, serological tests, which show past infection by checking blood samples for the presence of antibodies, have been around for a long time. The most commonly used test for this is the Indirect Fluorescence Antibody Test (IFAT), but Enzyme-Linked ImmunoSorbent Assay (ELISA) test kits have recently been proposed as a promising alternative. Serological tests are generally more suitable for epidemiological studies and screening for malaria at bloodbanks, than for diagnosis in clinical settings. However, under limited circumstances, they have been shown to also be suitable for diagnosis [39, 40].

1-2 Research objective

When comparing all diagnostic methods discussed, it becomes clear that no test is available yet that combines the simplicity and suitability for point-of-care testing of RDTs, with the excellent sensitivity and specificity of PCR and more novel DNA detection tests. Light microscopy remains a good middle ground between the two, while offering the added ability to determine infection stage and parasitaemia. However, interpreting the images is labour intensive, and its sensitivity and specificity are limited by the skill level of the microscopist. When developing an alternative diagnostic method that is suitable for point-of-care testing in resource-limited settings, it would ideally satisfy the following criteria:

Diagnostic Test Requirements

1. Detection limit ≤ 50 parasites / μl of blood, specificity $\geq 90\%$;
2. Ability to determine parasitaemia count;
3. Ability to identify parasite species and stage;
4. Low cost of equipment, minimal use of electricity, etc.;
5. Minimal labour and skill required.

It is clear that any malaria diagnosis method would become far more suitable for the target setting when the dependency on a human expert is limited or altogether removed. This can be achieved through automation. Introducing automation into the diagnostic procedure can vary from introducing automated stages to increase throughput, to automating the interpretation of the result. The focus of this thesis will be on the latter. Since light microscopy with Giemsa stained thin and thick blood smears is most significantly limited the time and expertise it takes to interpret the images, this is seen as a major area for potential improvement.

The aim of this work is to explore how recent advances in image classification technology can be used to automate the interpretation of blood films. By integrating this with the use of a low-cost, portable light microscope, we can develop a method that has the potential to meet all requirements listed above. We therefore specifically aim to develop a classification method that is suitable for the interpretation of images produced by the 'Assist B.02', a portable microscope that is currently in development at AiDx, which uses a $20\times$ objective instead of the standard $100\times$ oil immersed objective.

Thus, the main research question we attempt to answer is:

To what extent can neural networks be applied towards eliminating the need for trained experts in the interpretation of low magnification Giemsa stained thin blood smears for malaria diagnostics?

In order to investigate this, a two step image classification strategy is proposed; in which the erythrocytes are first segmented from the blood smear images to compute a cell count and then classified as either being either healthy or infected with Plasmodium parasite to compute an infection rate.

This thesis is structured as follows. Firstly, in chapter 2, background for this work is provided. Previous work on the subject of automated image analysis and its application to malaria diagnostics is discussed to identify which of the techniques that have been applied previously to this problem are promising, and which strategies have not yet been explored. This provides a motivation for the choice of neural networks, the concept of which is introduced more in-depth, to provide a theoretical framework for the work that follows. In chapter 3, the proposed image analysis methods are introduced, as well as the blood smear image datasets that were used to develop and test these methods. Two different erythrocyte segmentation methods are presented; a basic threshold based method, and a method based on a U-shaped convolutional neural network architecture. A classification method based on transfer learning, which exploits the existing ‘VGG-16’ neural network architecture is proposed. In chapter 4, the experimental results of applying these methods are presented, and performance measures are calculated. In chapter 5, a discussion on the results and methods used is provided; limitations are discussed and recommendations for future work are made. Finally, in chapter 6, conclusions are presented.

Chapter 2

Background

In this chapter, a review of previous work on the application of automated image analysis techniques to the diagnosis of malaria is presented. Automated image analysis is a varied field of research in which numerous techniques have been developed, of which the applications are constantly expanding. The field has undergone explosive growth in recent years, due to the rapid development of machine learning computer vision techniques.

A lot of research has been published on applying these techniques to the classification of Giemsa stained thick and thin slide microscopic images, to aid in the diagnostic decision making process [4]. Several classifiers, which are algorithms in which objects are divided over classes to minimise intra-class variance and maximise inter-class variance based on their features, have been proposed to automatically localise erythrocytes and determine cell count, as well as determine whether these erythrocytes are infected with Plasmodium parasite.

Most proposed classification algorithms follow the same basic steps; preprocessing, segmentation of erythrocytes, feature extraction and erythrocyte classification. These steps, as well as techniques that have been proposed for each of them, are described in section 2-1. A discussion of literature in which this conventional set of steps is followed is provided.

The recent development of ‘deep learning’, which refers to the use of artificial neural networks with multiple hidden layers as classifier, is identified as a high potential technique for the purpose of image classification in general and our objective in specific. The mathematical principles behind these classifiers are introduced in section 2-2, to provide a theoretical framework for the work that follows in this thesis. Previous work on the application of deep learning to malaria diagnosis is discussed, and limitations to this research are identified, which serve as a starting point for the method that will be proposed in the following chapter.

2-1 Conventional malaria image analysis techniques

Most algorithms proposed in literature are focussed on the classification of thin-smear Giemsa stained images, acquired through the procedure described in section 1-1-1. They aim to

automatically count all uninfected and parasitised erythrocytes, and often follow the following steps to do so; (1) preprocessing the blood smear image, (2) segmenting the erythrocytes from the background, (3) extracting parasite features and (4) mathematically dividing the erythrocytes into classes. This approach is schematically depicted in figure 2-1. Examples of techniques used for each step are given below.

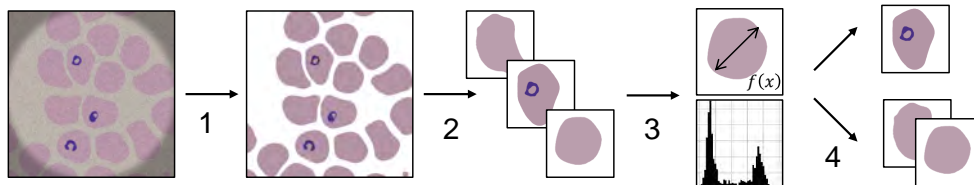


Figure 2-1: Schematic representation of the basic image analysis pipeline followed by most (traditional) automated malaria diagnosis algorithms, the numbers underneath the arrows refer to the four operations in this pipeline; **1)** preprocessing, **2)** segmentation, **3)** feature extraction and **4)** classification.

1. Preprocessing

Preprocessing is aimed at removing noise and enhancing image quality, and is often the first step when performing digital analysis on any type of image data. For noise removal, lots of established filters exist, such as median or Gaussian. In median filters, each pixel value is simply replaced by the median of those in a radius surrounding it. In Gaussian filters, a Gaussian distribution function in two dimensions is used to determine a weighted average of each pixel's neighbourhood, which then replaces that pixel value. These basic filters remove noise sufficiently and are often implemented in proposed automated malaria diagnostic systems, though more complicated filtering techniques have also been used. [41, 42, 43, 44, 4].

Low contrast is also a common problem, which is most commonly fixed through contrast stretching or histogram equalisation techniques. Contrast stretching is a linear normalisation that stretches the interval of the intensities of an image to a larger target interval. Histogram equalisation is a non-linear normalisation that stretches the histogram areas where intensities are concentrated and compresses the area with low abundance intensities [45, 41, 43].

Other problems that are typical for Giemsa stained thick- and thin film microscopic images are uneven illumination and variations in staining color. This can be fixed through color normalisation techniques, one that is often used is gray world assumption[46, 47].

2. Erythrocyte segmentation

When the thin smear is of good quality, meaning cells are separated completely and the image is in focus and well-illuminated, segmenting the individual erythrocytes is fairly straight-forward. It can be achieved through basic thresholding techniques, such as Otsu's, which optimally divides pixel values into two bins. This works well when the image is strongly bimodal, which can partly be achieved through preprocessing [48, 49]. When bimodality can't be achieved through preprocessing, or when the image is blurred, K-means clustering is a good alternative to iteratively assign pixels to foreground or background. Its disadvantage is that is more computationally complex than

thresholding techniques [50].

Problems with both methods arise when cells are touching or overlapping. To separate individual erythrocytes when this is the case, many methods have been proposed. Some simply iteratively threshold the larger objects until only objects that are the approximate correct size remain. This method can work well under circumstances, but is not very robust [51]. Watershedding is also a popular algorithm for cell segmentation, but its success is heavily dependent on the boundary gradients of the objects not being too weak [52]. Circle Hough Transforms have also been used and can work well, but they assume a circular shape and fixed size for the erythrocytes and fail when cells deviate from these assumptions too much [53].

3. Feature extraction

In pattern recognition, feature extraction refers to computing values from the raw (pixel) data that will optimally provide information for the classification that you want to perform, without loss of information or redundancies.

For diagnosis of blood slides with stained parasites, colour values of pixels are obviously informative features for determining infection. From these, features such as co-occurrence matrices, local binary patterns and histogram of oriented gradients can be computed. Some papers have proposed specifically extracting colour features only from the green channel of an image in RGB-colour space, as it provides the most contrast between the erythrocyte and the stained parasite. Others have suggested transforming the image to HSB-space before extracting colour features, or using a combination of both.

Morphological features, such as granulometry and relative shape measurements, can also be computed to aid in classification [54].

4. Classification

When dividing objects over classes, the objective is to minimise inter-class variance, based on the object features supplied. Essentially, a classification algorithm approximates a mapping f from the input features \mathbf{x} to the output class y , such that $\hat{f}(\mathbf{x}) \approx y$. An example of a simple classification method is the earlier mentioned ‘thresholding’, where objects are divided into classes based on whether their value is above or below a certain threshold. More complicated classification methods often use a training set of pre-classified objects to find a classification strategy that minimises the error rate, which is called ‘supervised learning’. ‘Unsupervised learning’, where only the input data and the cost function are known a priori, is also possible. A great number of learning algorithms have been developed, such as Support Vector Machine (SVM), Bayesian classifiers, K-nearest neighbour classifiers, logistic regression trees, artificial neural networks and many more. All of these have been applied to analysis of malaria infected thin smears. Binary classification, dividing objects simply into non-infected and parasitised erythrocytes is a common objective, but attempts have also been made to further divide parasitised cells in up to 20 classes (one for each life stage of each species, as described in table 1-1), for example by Tek et al [55]. Success of these methods is dependent on the quality and the separability of the features that were extracted from the erythrocytes and the parasites.

A typical example of a method following this pipeline was proposed by Savkare et al. They collected RGB images and preprocessed with median and Laplacian filters and standard his-

togram equalisation. They converted the image to grayscale, and applied Otsu thresholding to the grayscale image and to the green channel, resulting in two separate binary masks which were combined into one. The average size of erythrocytes was calculated, objects not corresponding to this size were deemed artefacts or leukocytes and removed, resulting in a binary mask of background and erythrocytes objects. Success rate of 99.43 % was reported in the recognition of erythrocytes, however, in calculating this rate, objects consisting of multiple erythrocytes were deemed correctly recognised. Watershedding was applied to segment these objects and resultant objects too small to be erythrocytes were removed. Given the high accuracy reported before the watershedding, the removed objects were likely the result of over-segmentation, but since no separate accuracy is reported after this step, it is impossible to tell how many separated erythrocytes were successfully found with this method.

Of the resultant objects, mean, standard deviation and third moment of the green channel histogram were calculated, as well as shape and textural features. These were used to determine whether erythrocytes were infected using a SVM, sensitivity and specificity for this classification were reported at 93.12 % and 93.17 % respectively [56].

Another example is the method proposed by Das. et al., who investigated the classification of thin blood smears infected with *P. vivax* and *P. falciparum*. They preprocessed their images using gray world assumption to correct illumination and geometric mean filter to remove noise, before applying marker controlled watershedding to segment erythrocytes. No separate performance measures were reported for the segmentation. They then computed a total of 96 textural and morphological features, the most significant of which were selected through statistical analysis. After this a Bayesian classifier and a SVM were trained to classify erythrocytes, not only as infected or non-infected, but also to distinguish between 5 *P. vivax* and *P. falciparum* life-stages. On this task, an accuracy of 84 % was obtained with the Bayesian classifier [57].

The above reported performance measures are comparable to those of other methods that use different techniques following the same basic pipeline. Das et al. compiled an extensive review the literature on this topic, and compared accuracy and (where given) sensitivity and specificity scores of 35 different published approaches, which were typically in the range 80 – 100% [58]. However, the images used to test each of these methods is different, so it is impossible to fairly compare between them on the basis of the reported performance measures.

2-2 Neural Networks and malaria image analysis

As stated previously, for malaria image classification, no standardised comparison is currently available, making it very hard to quantify the state-of-the-art. However, in more general image classification research, such comparisons are possible. Image classification contests such as the ImageNet Large Scale Visual Recognition Challenge (INLSVRC), make it very clear that the field has become dominated over recent years by deep learning techniques which use Artificial Neural Network (ANN) [59]. Therefore, these will be discussed more in-depth here.

2-2-1 Mathematical principles

Basic principle

An ANN is a type of classifier, inspired by biological neural networks, in which the feature extraction and classification are combined in one algorithm. The most basic type of ANN is a feedforward neural network, or multilayer perceptron, which is schematically depicted in figure 2-2.

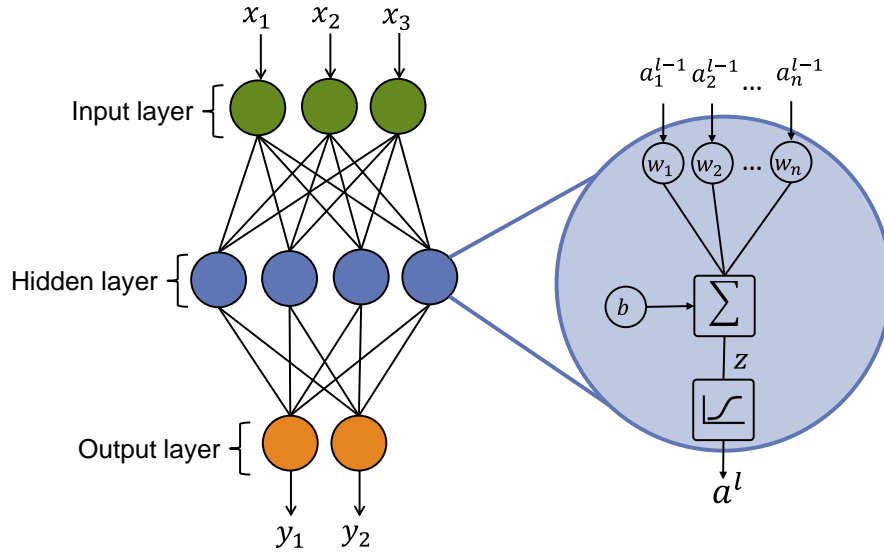


Figure 2-2: Schematic depiction of a feed forward neural network with three inputs, two outputs and one hidden layer. On the right side, the general architecture of a single neuron is depicted.

They consist of an input layer with all the data input points, an output layer in which inputs are mapped to outputs, and (optionally) any number of hidden layers. If all nodes in all layers pass outputs to each other, like in the network depicted here, the ANN is referred to as ‘fully connected’. When many hidden layers are incorporated into the architecture of the network, they are often referred to as ‘deep neural networks’ and the training and application of the network are called ‘deep learning’. The hidden layers consist of artificial neurons. In each of these neurons a combination of an affine transformation and a non-linear activation function are used to transform the inputs. Let the output of a single neuron k in layer l , be denoted a_k^l . Each neuron uses the vector of outputs of the previous layer \mathbf{a}^{l-1} as inputs, the first step is to compute a weighted sum z_k^l of these;

$$z_k^l = \sum_{i=1}^n (a_i^{l-1} w_{ki}^l) \quad (2-1)$$

where n is the dimension of the previous layer, $w_{k1} \dots w_{kn}$ are weights of the neuron. A bias b_k is added, and the output is then computed by applying some non-linear activation function g ;

$$a_k^l = g(z_k^l + b_k^l) \quad (2-2)$$

This output is then propagated to the neurons in the next layer, where they the same type of transformation. The total mapping of the inputs \mathbf{x} to outputs \mathbf{y} is thus a function of all the

weights and biases; $\hat{\mathbf{y}} = f(\mathbf{x}, \mathbf{W}, \mathbf{b})$. The correct mapping from the inputs to the outputs is approximated such that $\hat{\mathbf{y}} \approx \mathbf{y}$ by adjusting the weights and biases during learning. Neural networks have been proven to be universal function approximators, meaning that any mapping can be approximated arbitrarily well, given enough hidden units are used [60].

Activation functions

The activation function plays an important role in the approximating ability, without them, only linear mappings could be approximated. Different commonly used activation functions are shown in figure 2-3.

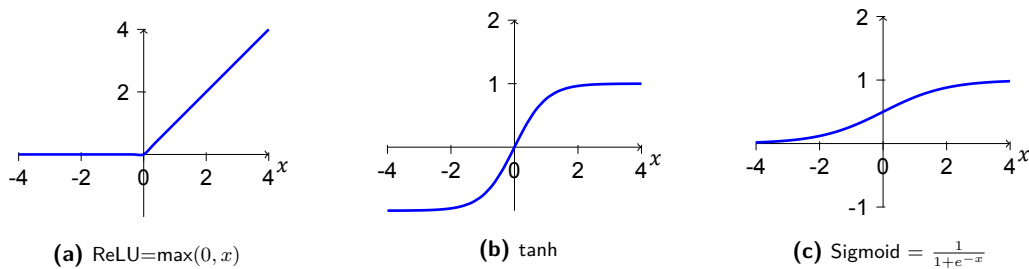


Figure 2-3: Different activation functions used by artificial neurons.

The choice for which activation function to use is an important design parameter. Sigmoid and tanh both have smooth gradients and normalise the outputs of the neuron. As opposed to the sigmoid function, tanh is zero-centered, which makes it more suitable for inputs that can be strongly negative. The disadvantage of using these S-shaped activation function, is that their gradients become very small for large values, which can slow down the learning of the network significantly. This has been termed the vanishing gradient problem. Furthermore, both activation functions are fairly computationally demanding. This is why, Rectified Linear Unit (ReLU), which is a piecewise linear function that outputs the input value when it is positive, and 0 otherwise, has become a popular choice. Neurons that use a ReLU activation function can become inactive when only negative values are put through, resulting in sparsity in networks, which can be advantageous in training [61]. However, when too many units become inactive, this impedes learning, which is referred to as the ‘dying ReLU problem’. Several suggestions have been made to prevent this, such as using ‘Leaky ReLU’, which passes a scaled down output for negative values ($\max\{0.1x, x\}$) [62].

Convolutional Layers

A Convolutional Neural Network (CNN) is a type deep neural network that was developed specifically with the goal of image classification in mind. A core concept in the architecture of CNNs is the introduction of convolutional layers. Unlike in the previously described fully connected layers, in convolutional layers, the input of each neuron is a function of only a small region of the outputs of the previous layer. This input is produced by convolving the previous layer with a small matrix of weights called a kernel. The kernel ‘slides’ over the

original image, and the convolution of the kernel with the region surrounding the input pixel is computed, by;

$$z_{ij} = W * x_{ij} = \sum_a \sum_b w_{ab} \cdot x_{(i-a)(j-b)} \quad (2-3)$$

where $w_{ab} \in w_{00} \dots w_{NN}$ are the weights in the kernel W of size $N \times N$ and $x_{ij} \in x_{00} \dots x_{nn}$ are the values of input matrix X with size $n \times n$. The convolution z_{ij} is then passed through an activation function, to produce the output y_{ij} ;

$$y_{ij} = g(z_{ij} + b) \quad (2-4)$$

Equations 2-3 and 2-4 replace equations 2-1 and 2-2. Besides this, the convolutional layer are implemented in the same way as the standard network layers described above.

Note that the neurons in convolutional layers are structured in a grid, this make convolutional layers especially suitable for the classification of structured data such as image data. The kernel essentially acts as a feature extraction filter, where the learnable weights converge towards features in the image. By using the same kernel with the same weights on the entirety of the input, an activation map of these features is produced. The output of the convolutional layer is therefore called a feature map.

The convolutional layer operation is schematically depicted in figure 2-4.

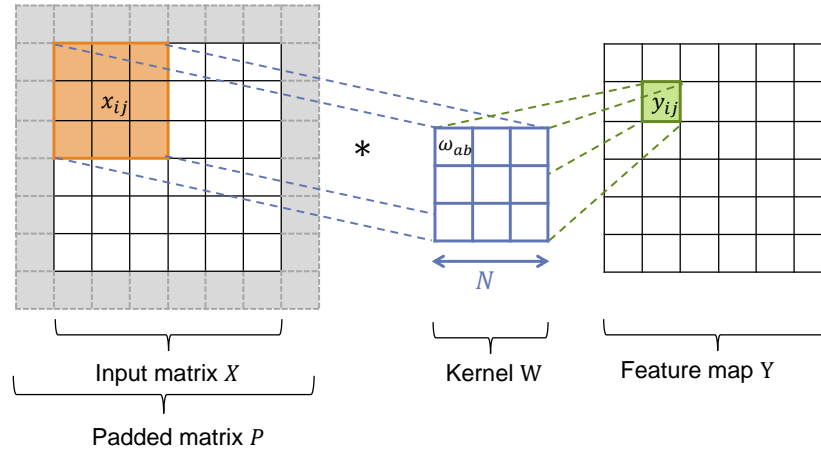


Figure 2-4: Schematic depiction of the convolution of a 6×6 input image with a 3×3 kernel. In order to produce a 6×6 feature map, padding is used.

Given an $n \times n$ image X as input, and a $N \times N$ kernel W , which slides over the input matrix with stride 1 (meaning it moves 1 pixel for each convolution), the size of the feature map will be $n - N + 1 \times n - N + 1$. When a feature map of equal size to the input is desired, padding can be used around the input matrix. This is also depicted in figure 2-4. Often, multiple kernels are used in one convolutional layer to produce multiple feature maps. If M kernels are used, the size of the output (with padding) will be $n \times n \times M$.

The convolution described above assumes a single channel input. It is possible to have a multi-channel input to a convolutional layer. In this case, the convolution can be described as;

$$z_{ij} = W * x_{ij}^k = \sum_{k=1}^K \left(\sum_a \sum_b w_{ab}^k \cdot x_{(i-a)(j-b)}^k \right) \quad (2-5)$$

Here, x_{ij}^k refer to the pixels in the k th input channel, the total number of input channels is K . The kernel in this case takes the size $N \times N \times K$, but the output remains two dimensional. Even though the kernel is now 3D, this is still referred to as a 2D convolution, since the kernel slides over the input only in horizontal and vertical direction. It can be thought of as a stack of filters, where each filter is convolved with one input channel, and the outputs of the convolution are summed.

In order to reduce the dimensionality and prevent over-fitting in CNNs, pooling layers are often added after convolutional layers. In these, the outputs of the convolutional layers are down-sampled. The $n \times n$ feature map is reduced in size to $\frac{n}{p} \times \frac{n}{p}$, by dividing the feature map in $p \times p$ patches and taking some function of the values in this patch as the output. In average pooling layers, the average of the values is passed, while Max pooling layers pass the largest value.

It is also possible to up-sample through convolution, when a feature map of a bigger size than the input is desired. This concept was introduced as ‘deconvolution’, but ‘transposed convolution’ has since been suggested to be a more accurate name [63, 64]. To understand the concept of transposed convolution, first note that the convolution operation can be written as a matrix multiplication, by rearranging the weights of the kernel into a convolution matrix which represents all positions the kernel takes on the input, and rearranging the input matrix into a vector. This is explained visually for the convolution of an input image X of 3×3 with a 2×2 kernel W in figure 2-5. Now note that if the transpose of the convolution matrix is taken instead to produce the feature map of an image Z of size 2×2 , so $\tilde{W}^T \times \tilde{Z} = Y$, this feature map will be of size 4×4 , and thus has been up-sampled by the size of the kernel.

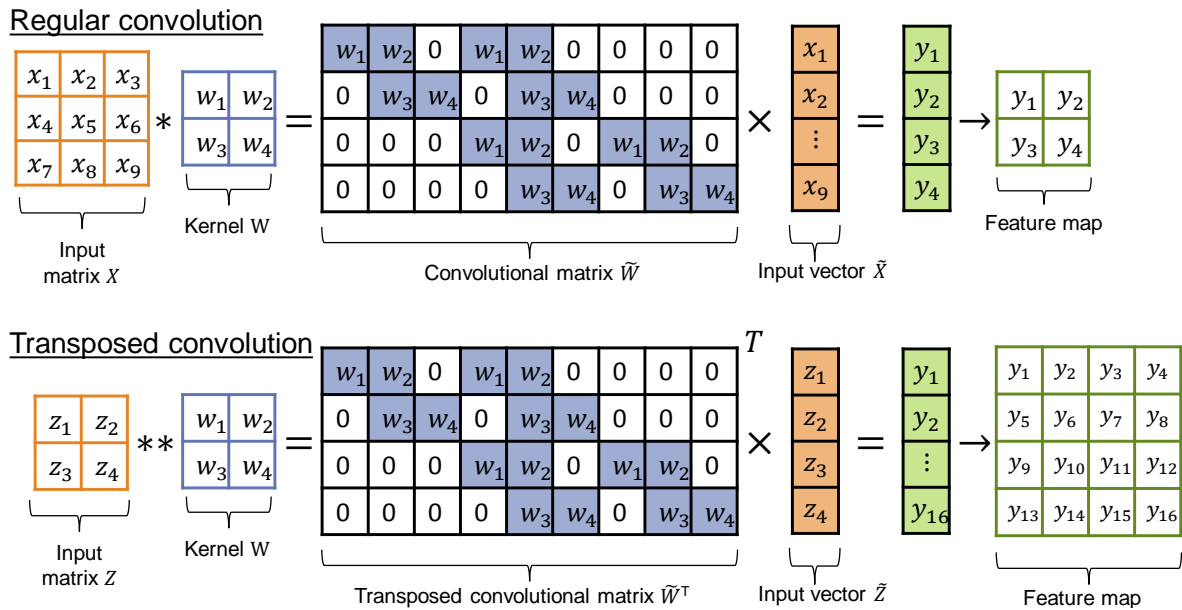


Figure 2-5: Upper image: convolution of a 3×3 input matrix with a 2×2 kernel to create a 2×2 feature map, expressed as a matrix operation. Lower image: transposed convolution of a 2×2 input image with that kernel, to create a 4×4 feature map, expressed as a matrix operation.

Training

The first step in training the ANN is *initialisation*. In order to ensure convergence of the network it is important that the outputs of layers don't explode or vanish after the first pass. Initialising the weights and biases in such a way that the standard deviation of the activation outputs of each layer is normalised is a good way to prevent this. In order to achieve this, the 'Xavier initialisation' was proposed, where the weights of a layer are drawn from a uniform set, which is bounded between $\pm \frac{\sqrt{6}}{\sqrt{n_i+n_{i+1}}}$, where n_i refers to the number of incoming network connections, and n_{i+1} the number of outgoing connections [65]. This strategy works well for continuous activation functions that are symmetric about zero, such as tanh. For asymmetric functions such as ReLU, a initialisation dubbed the 'Kaiming initialisation', in which weights are randomly drawn from a standard normal distribution and scaled by $\frac{\sqrt{2}}{\sqrt{n_i}}$, was shown to lead to faster convergence [66]. Biases are usually initialised at zero.

Each training iteration of the network can be divided into three phases: forward propagation of the data, backward propagation and optimisation. During *forward propagation*, the prediction $\hat{\mathbf{y}}$ of the current network on the data is computed, by computing equation 2-2 for every neuron in every layer. This prediction is used to determine the value of the loss function J , which is some measure of the total error in the system. Often the Mean Squared Error (MSE) is used;

$$J = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2-6)$$

where \mathbf{y} is a vector containing the ground truth of the network. Other options for loss functions that are commonly used are the Root Mean Squared Error and the the Mean Absolute Error. These loss functions are effective when the targeted output is a continuous value. When dealing with classification, the target output is one of integer classes. In this case, cross-entropy is a more effective measure of the error in the system, and therefore often used as loss-function. When dealing with a two class classification problem, the binary cross-entropy loss function is given by;

$$J = -\frac{1}{n} \sum_{i=1}^n y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i) \quad (2-7)$$

It is possible to add additional terms to the loss function to influence the outcome, for example, a regularisation term that penalises large weights, which can help reduce over-fitting. When extra terms are added, the objective function is no longer only a function of the loss, and is therefore referred to as cost function.

The next phase is *backpropagation*, during which the gradient of the cost function is calculated. This is done by computing an error function δ^l at each layer, by taking the derivative of the cost function with respect to the weighted inputs \mathbf{z}^l ;

$$\delta^l = \frac{\partial J}{\partial \mathbf{z}^l} = \sum_k \frac{\partial J}{\partial a_k^l} \frac{\partial a_k^l}{\partial z_k^l} = \nabla_a J \circ g'(\mathbf{z}^l) \quad (2-8)$$

where $\nabla_a J$ is a vector of derivatives of J with respect to the components of \mathbf{a}^l . In the output layer L , these components are known ($\hat{\mathbf{y}} = \mathbf{a}^L$), making it easy to compute the error of the output layer. This error can then be propagated back through the network to compute the

error of each neuron. An equation for the error at a layer $l - 1$, in terms of its succeeding layer is given by

$$\delta^{l-1} = ((\mathbf{W}^l)^T \delta^l) \circ g'(\mathbf{z}^{l-1}) \quad (2-9)$$

This can be used to compute the errors all the way through the network efficiently. When errors are known, these can be used to compute the gradient of the network by realising that

$$\frac{\partial J}{\partial b_k^l} = \delta^l; \quad \frac{\partial J}{\partial w_{ki}^l} = a_k^{l-1} \delta^l \quad (2-10)$$

The gradient is finally used to update the weights and biases during *optimisation*. A gradient descent method is used for this; since the gradient gives the direction of the largest increase of the cost function, in order to minimise it, a step in the opposite direction of the gradient is taken:

$$w_{ki}^l \leftarrow w_{ki}^l - \alpha \frac{\partial J}{\partial w_{ki}^l} J; \quad b_k^l \leftarrow b_k^l - \alpha \frac{\partial J}{\partial b_k^l} J \quad (2-11)$$

The size of this step α is called the ‘learning rate’ and it is a tunable parameter in training the network.

Often, the training samples are divided into batches, and the gradient is determined for all training samples in the batch before updating the weights. The size of this batch is a hyperparameter of the network which can be tuned to achieve the desired performance. Small batch size leads to stochastic weight updates, while large batch size leads to slow learning. An epoch is defined as the number of iterations after which all training data has been passed through the network exactly once.

2-2-2 Neural networks applied to malaria image data

Some research has been published on the application of neural networks to the classification of Giemsa stained malaria-infected blood smears.

Dong et al. trained three different CNN architectures on small dataset of segmented erythrocyte objects, which they obtained through thresholding and then applying a Hough circle transform to blood slide images. They used this to create training and testing sets of equal size, both with 765 non-infected and 517 infected cells in them. No performance metrics for the segmentation were given. The existing LeNet-5, AlexNet and GoogLeNet architectures were trained on these images, and accuracies of 96.18%, 95.97% and 98.17 % were reported for each of the networks respectively. This was compared with a SVM trained on the same data, in a similar way as done by Das et al., which was described in section 2-1, which achieved an accuracy of 91.66 % on the same data [67].

Rajamaran et al. also worked on the classification of Giemsa stained thin films. They first segmented the erythrocytes from blood slide images, using another conventional cell segmentation algorithm as described in section 2-1. They produced a database 27,558 cell images with equal instances of parasitised and uninfected cells, which they made publicly available, and went on to develop a CNN based classifier for. They proposed a network architecture consisting of three blocks of two convolutional layers, the first of which containing a max pooling layer, the second with a average pooling layer and the third followed by directly by three fully connected layers. On the object level, they achieved sensitivity and specificity of

93.11 % and 95.12 % respectively. The performance of their proposed network architecture was later compared with the use of pre-existing network architectures such as VGG-16 and ResNet-50, and slightly outperformed by these [68].

Gopakumar et al. proposed training a network on a focus stack of RGB cell images, instead of just a single image per cell object. This was claimed to improve performance in distinguishing parasites from artefacts such as dust specks. Segmented cells were acquired with a two-stage threshold based method. Details on the specific architecture of the CNN used were not provided. A sensitivity and specificity of 96.98 % and 98.50 % were reported respectively. However, the estimated parasitaemia produced by their total proposed algorithm was not very close to the ground truth, at 173 % of the actual parasitaemia [69].

All research described so far combined a CNN based classifier, with a simple segmentation method. Erythrocyte segmentation with CNN is also possible [70]. Delgado-Ortet et al. applied this to the classification of thin smear images, using a network architecture where convolutional layers are followed by deconvolutional layers to create a segmentation mask for slide images. They combined this with a CNN with 8 convolutional layers to classify segmentation output, achieving a global accuracy of 93.72 % on segmentation over the test set and a specificity for malaria detection of 87.04 % [71].

Finally, Mahanian et al. proposed a method for the classification of Giemsa stained thick blood smear images, using the Caffe CNN architecture, which uses 5 convolutional layers followed by 3 fully connected layers ([72]), as a feature extractor. The candidate objects produced by the network are then used to train a logistic regression classifier, which divides them into parasites and non-parasites with a sensitivity of 91.6% and a specificity of 94.1% [73].

2-3 Discussion of automated malaria diagnosis techniques

The methods discussed in this chapter, and their performance measures, are summarised in table 2-1. Making an objective statement on the relative performance of automated malaria image analysis techniques is difficult, since performance measures are usually only reported on a small set of (private) data. Often, no separate performance is reported for the segmentation of the erythrocytes, and the performance is only evaluated in terms of segmented cells classified correctly, making it impossible to assess to overall performance of the proposed method. Reporting the specificity and sensitivity on object-level only makes sense from a image classification point of view, but from a clinical point of view, these performance metrics are not very informative; number of cells identified correctly over an entire dataset, doesn't give insight into whether the classification is suitable for diagnosis at patient-level.

Furthermore, a limited number of images is often used for testing, which are typically acquired in exactly the same manner as the images used to develop and train the algorithm were. Especially in conventional image analysis techniques, the extracted features that are used, such as size parameters and staining colours, can vary heavily if the images are acquired with a different method or even just a different camera, it is doubtful these methods will perform well when tested on images from another source.

In practice, smear and image quality can be of much lower quality than they are under ideal lab conditions, but research that deals with the classification sub-standard microscopic images has thus far been limited.

Table 2-1: Summary of automated malaria classification methods. Performance is given in terms of accuracy (acc) or sensitivity (sens) and specificity (spec), depending on what was reported.

Ref.	Segmentation method	Performance	Classification method	Performance
[56]	Otsu thresholding + watershed	Acc 0.994 (b. watershed)	SVM	Sens 0.931, Spec 0.932
[57]	Marker controlled watershed	-	SVM with feature selection, classification of species/stage	Acc 0.84
[67]	Threshold + Hough circle transform	-	CNN: GoogLeNet	Acc 0.982
[68]	Level-set based algorithm	Sens 0.962 PPV 0.944	Custom CNN	Sens 0.931, Spec 0.951
[69]	Two stage threshold	-	CNN	Sens 0.970, Spec 0.985
[71]	Convolutional + Deconvolutional NN	Acc 0.937	Custom CNN	Spec 0.87
[73]	Thick smears used, no segmentation	-	CNN + logistic regression	Sens 0.916, Spec 0.941

In general, it is clear that for an automated classification technique to be suitable for use in diagnostics, its performance would ideally be just as high or higher than that of a human expert. We can define the performance of a human expert by looking at the requirements the World Health Organization (WHO) sets on a ‘level 1’ microscopist; which are given in table 2-2.

Table 2-2: Performance requirements for WHO microscopist competence levels, from [5].

Competence Level	Parasite detection (%)	Species identification (%)	Parasite count within 25% of true count (%)
1	90-100	90-100	50-100
2	80-89	80-89	40-49
3	70-79	70-79	30-39
4	0-69	0-69	0-69

In terms of parasite detection, the techniques discussed in this section are generally claimed to perform above 90%, so as good as a level 1 microscopist. Automated species classification has also been attempted, results thus far have not been as good as those of classifiers that only determine infection. Even though reported sensitivities and specificities are high, the methods reported don’t necessarily result in accurate parasiteamia counts; Gopakumar et al. reported the highest performance measures of all methods discussed, but their parasiteamia was 73 % off [69].

In the general field of image analysis, CNN based deep learning techniques have shown impressive performance on image classification, and the research published so far on the application of them for malaria diagnosis is promising. This research is however limited; it is mostly focussed on the classification of pre-segmented thin smear erythrocytes, which means only part of the diagnostic process is performed by the algorithm. Accurate cell segmentation with help of CNN has for this specific problem, been scarcely attempted. Furthermore, no attempts were found to verify the performance of a classifier trained on a large set of erythrocyte images, as was done by Rajaraman et al., on image data that was acquired with a different set-up.

Chapter 3

Method

In this chapter, an automated method for interpretation of Giemsa stained thin smears is proposed. The process of classifying Giemsa stained thin smears was split into two steps; first the erythrocytes are segmented out of the full blood slide image, and then the individual erythrocytes are classified. The first stage is needed to determine cell count and create classifiable objects for the second stage. It is theoretically possible to detect malaria parasites without first segmenting out the individual erythrocytes, however, this approach is far more demanding for the classifier and would make it impossible to also determine parasitemia, so was not chosen here.

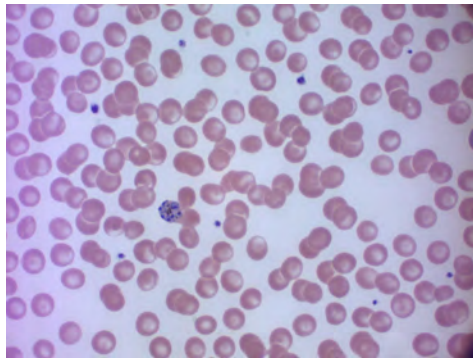
Firstly, the datasets that were used in the development and testing of the algorithm are discussed in section 3-1. Then, two different methods for erythrocytes segmentation are presented. A simple threshold based segmentation method was implemented, which is described in 3-2. A more sophisticated classifier, based on a ‘U-shaped’ Convolutional Neural Network (CNN) architecture was also developed, which is described in section 3-3. Performance of both methods will be compared in chapter 4, and the segmentation method that produces the best results will be included in the full data interpretation pipeline.

Finally, in section 3-4, the concept of ‘transfer learning’ is utilised to design a classifier based on the existing VGG-16 CNN architecture.

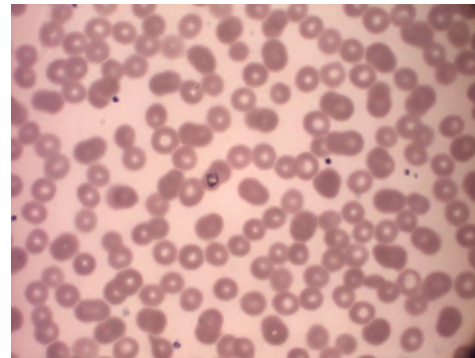
3-1 Datasets

In developing and testing the segmentation and classification methods described in this chapter, three datasets of Giemsa stained thin blood smears infected with *P. Falciparum* were used. The first one, which was used for segmentation, consists of 18 blood film images, taken from the database that was made available by Loddo et al. for public use [74]. This data-set is used to develop and test the segmentation algorithm. We will refer to these images as the ‘*Loddo dataset*’ from now on. The blood films were magnified with a 100× oil immersion objective, and pictured using a Leica DM2000 optical laboratory microscope, with a 30 Watt halogen light source and a built-in 5 megapixel camera. Every image is stored in PNG format

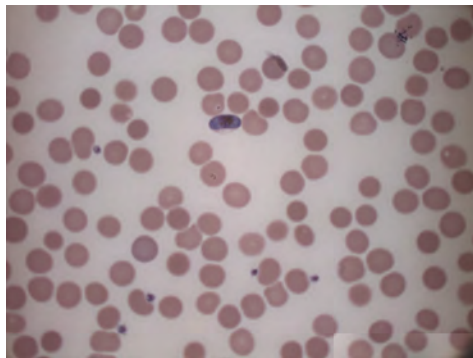
with a 2592×1944 resolution and 24 bit colour depth. The Field of View (FOV) of each image is about $140\mu m \times 100\mu m$. Figure 3-1 shows a sample of this dataset. As can be seen, the quality of the thin smears and the illumination varies over the images; some of them have large amount of overlapping cells. This is realistic to images taken in practical settings, and makes this dataset suitable for testing the robustness of a segmentation method with respect to these differences.



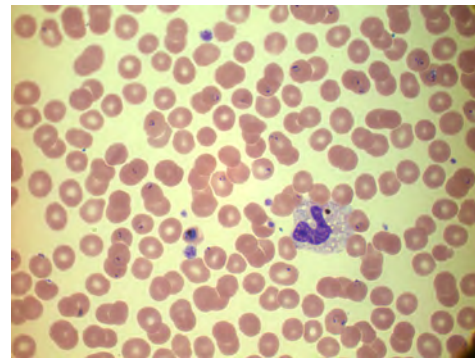
(a) Well contrasted, cells mostly separated with central pallor visible.



(b) Poorer contrast and focus, groups of overlapping cells present.



(c) Cells very well separated, no central pallor visible.



(d) Good contrast, some overlap, one white blood cell visible.

Figure 3-1: Four images from the Loddo dataset of Giemsa stained thin blood smears infected with *P. falciparum*, taken with $100 \times$ oil immersed objective.

To develop and test the classification algorithm, a dataset of segmented erythrocytes was used, which was made available for public use by Rajaraman et al. [68]. This dataset will be referred to as the '*Rajaraman dataset*'. To create it 200 Giemsa-stained thin blood smears, 150 *P. falciparum* infected and 50 healthy, were photographed with a smartphone camera attached to a conventional light microscope with a $100\times$ oil immersed objective. Erythrocytes were cropped out of these images with a level set algorithm, and hand-labelled as either parasitised or uninfected. In total, the dataset consists of a total of 27,558 cell images with equal instances of both classes. Cells were cropped at the cell border and the images were stored as three-channel RGB images in PNG format. Cells were not resized or reshaped, so the dimensions vary over the dataset, average size of one image is $\pm 120 \times 120$ pixels. A sample of this dataset is shown in figure 3-2.

The final dataset used contains six images, which were taken with a digital microscope that

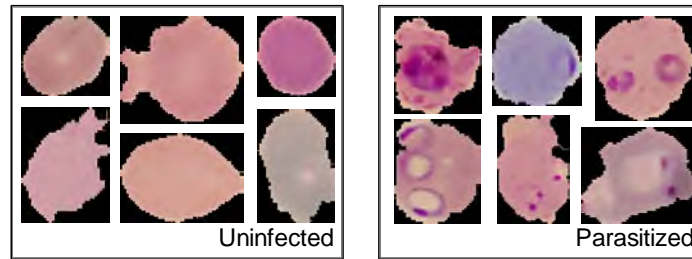


Figure 3-2: Six images from each of the two classes in the Rajaraman dataset

is currently in development at AiDx called the Assist B.02. We will refer to these images as the ‘*AiDx dataset*’ from now on. The blood films were magnified with a $20\times$ lens and illuminated with a white LED (wavelengths $400nm - 650nm$). The images were captured with a 18 megapixel color camera and stored in JPEG format. The average FOV of the images is $250\mu m \times 190\mu m$. This larger FOV is an obvious advantage compared with conventional thin smear microscopic images, however, the lower magnification that was used to achieve it, causes some detail to be lost in the images, making segmentation and classification more challenging. Figure 3-3 shows a sample from this dataset.

Infected cells in these blood film images were manually annotated and verified by an expert. This dataset is used to test both the segmentation algorithms and the classifier and gain insight on the performance when they are implemented back-to-back. It was chosen for this specifically because the AiDx microscope was developed for in situ use, which is also the target setting for the method proposed here.

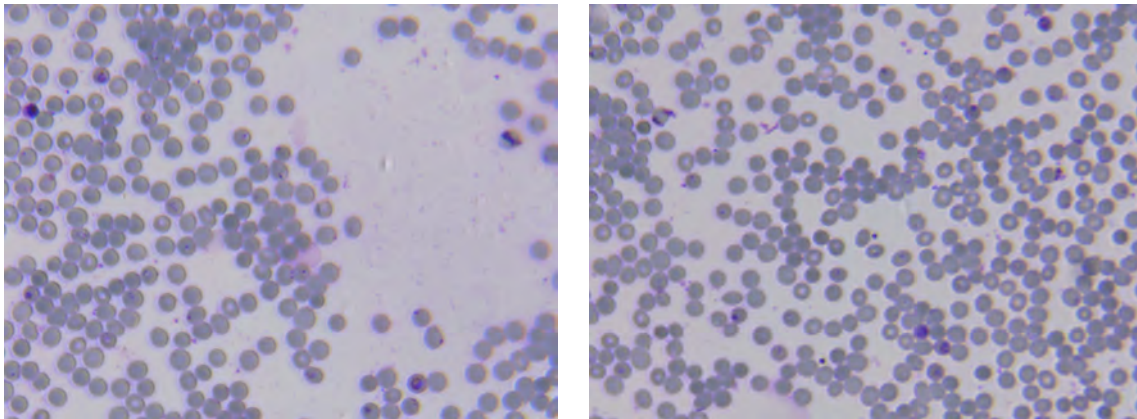


Figure 3-3: Two images from the AidX dataset of Giemsa stained thin blood smears infected with *P. Falciparum*, taken with $20\times$ objective.

3-2 Threshold based segmentation

The first method used for segmentation uses an automatic thresholding method to divide pixels into background and erythrocytes. This method was implemented using the open-source ImageJ software [75]. The steps of the algorithm are schematically depicted in figure 3-4.

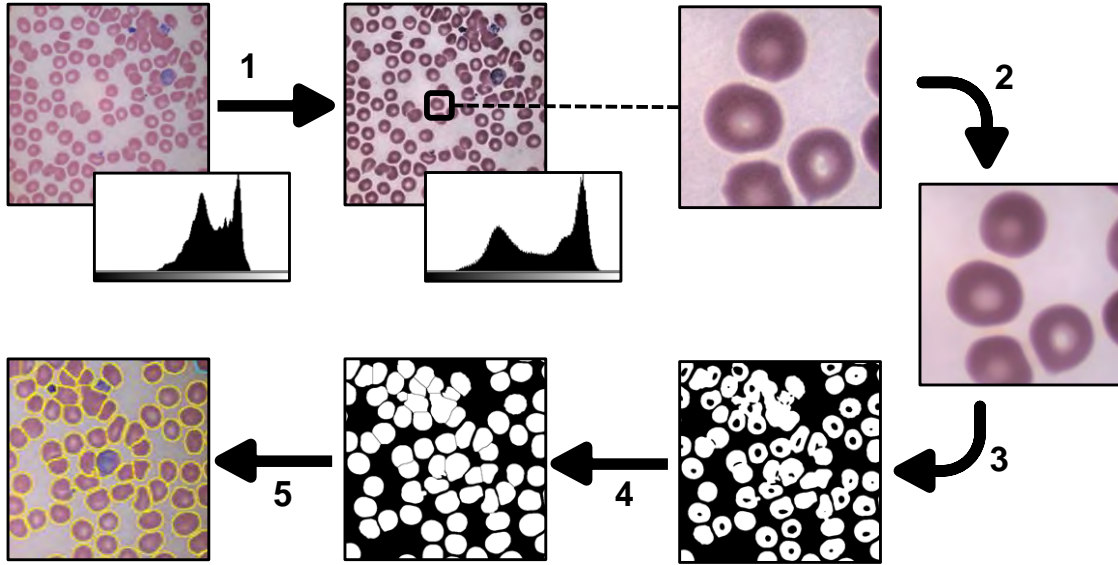


Figure 3-4: Schematic depiction of the segmentation algorithm, which pre-processes the blood film images (step 1-2), divides them into binary classes (step 3), separates the resultant objects through watershedding (step 4) and crops them out of the original image (step 4-5).

1. Contrast Limited Adaptive Histogram Equalization (CLAHE)

In order to increase the contrast between the erythrocytes and the background, a modified histogram equalisation called CLAHE is applied [76]. Histogram equalisation can be used to increase the contrast in images, by distributing the intensities evenly over the histogram bins. This is done for each of the channels in the image separately. Given a three-channel RGB image, one channel f contains a total of N pixels with L possible integer intensities, the total number of pixels with an intensity n is denoted N_n . The normalised histogram bins p_n are defined by;

$$p_n = \frac{N_n}{N} \quad n = 0, 1, \dots, L - 1 \quad (3-1)$$

When applying standard histogram equalisation, the intensity of each of the pixels in f (denoted $f(i, j)$) are transformed by multiplying with the cumulative distribution function of that intensity. The equalised image channel g is defined by;

$$g(i, j) = \text{floor}(L - 1) \sum_{n=0}^{f(i,j)} p_n \quad (3-2)$$

In the images we wish to process, illumination is often uneven, resulting in a non homogeneous distribution of pixel intensities over the image. When the same transformation is applied to the entire image, this uneven distribution is only further amplified, and the contrast between objects and background is not increased in all regions. This uneven illumination can be fixed through adaptive histogram equalisation, where instead of using the histogram of the full image to transform each pixel, the histogram is calculated

for a neighbourhood region of the pixel, the size s_H of which is a tunable parameter. The intensities in a region are distributed over the entire possible range when taken this approach, thus maximising the contrast, which is not always desirable. When a pixel has a fairly homogeneous neighbourhood, because the entire region consists of background pixels, implementing standard adaptive histogram equalisation would cause the noise in those regions to be over-amplified. For this reason, a contrast limit was added, which puts a maximum on the slope that the intensity transfer function (eq. 3-2) is allowed to take. This contrast limit β is also a tunable parameter.

2. Median Filter

In order to filter out the noise, a 2D-median filter is applied to each of the channels. This filter replaces each pixel value with the median of the surrounding pixel values, within a neighbourhood of size s_M . So for pixel $f(i, j)$, the transformed pixel is given by;

$$g(i, j) = \text{median} \left(\begin{bmatrix} h(i - s_M, j - s_M) & \dots & h(i + s_M, j - s_M) \\ \vdots & \ddots & \vdots \\ h(i - s_M, j + s_M) & \vdots & h(i + s_M, j + s_M) \end{bmatrix} \right), \quad (3-3)$$

3. Otsu Thresholding

The pixels in the image are first converted to grayscale, and then divided over two classes; background and foreground. When divided correctly, all erythrocytes will be in the foreground. The classes are defined by a threshold t , every pixel with intensity $n = [0, t]$ is assigned to class 1, and every pixel with intensity $n = [t + 1, L - 1]$ is assigned to class 2. Otsu's method is used to find an optimal threshold t^* , that minimises the intra-class variance, which is the sum of the variances of the classes, weighted by their cumulative probability density function [77]. For two classes, this is equivalent to maximising the inter-class variance, which can be written as ¹

$$\sigma^2(t) = \frac{[\mu_f \omega(t) - \mu(k)]^2}{\omega(t) [1 - \omega(t)]} \quad (3-4)$$

where $\omega(t)$ is the cumulative probability density at the threshold;

$$\omega(t) = \sum_{n=0}^t p_n \quad (3-5)$$

$\mu(t)$ is the mean of the histogram up to the threshold;

$$\mu(t) = \sum_{n=0}^t n p_n \quad (3-6)$$

and μ_f is the mean intensity of all the pictures in the image, which is equal to $\mu(L - 1)$. The optimal threshold is found through exhaustive search, i.e. calculating $\sigma^2(t)$ for every $t \in [0, L - 1]$, and then setting t^* such that

$$\sigma^2(t^*) = \max_{0 \leq n < L-1} \sigma^2(t) \quad (3-7)$$

¹This equation only holds for a two-class problem. The full derivation of this equation for the inter-class variance can be found in [77].

4. Binary Operations & Watershedding

Because the hemoglobin in erythrocytes concentrates at the cell boundaries, their centres are lighter when viewed under a microscope (this is called central pallor). When converting the blood film images to binary images, this causes holes to be present in the foreground. These holes are filled with binary operations; first some ‘dilations’ are applied, meaning pixels are added to the edges of the objects, followed by some ‘erosions’, which remove pixels from the edges of the objects. This encloses any open areas present in the objects. The number of dilations and erosions needed to get good results, differs with the magnification, sharpness and resolution of the original image, so this is a tunable parameter of the algorithm called ν . After this, holes are filled, meaning every background pixel fully enclosed by foreground objects is converted to foreground. This operation can either increase or decrease the performance in the next step, so including it is optional.

After this, touching foreground objects are separated with the ‘watershed’ method [78]. In order to do so, the euclidean distance map of the binary image is calculated, and the ‘ultimate eroded points’, which are the local maxima of this distance map, are found. These points are then iteratively dilated until the edge of the object is reached, or the edge of another of the region of another point. This process is visualised in figure 3-5.

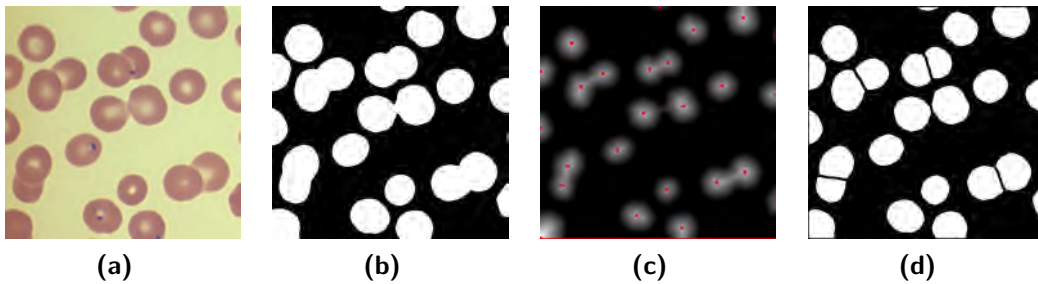


Figure 3-5: Visualisation of the watershedding algorithm; **a)** shows the original thin blood film with several overlapping cells, **b)** shows the binary mask based on this image, **c)** shows the euclidean distance map of this binary image, with the ultimate eroded points, and **d)** shows the resultant binary mask after watershedding.

5. Transfer mask

Finally, all foreground objects in the binary image are measured. Erythrocytes have a diameter of $6 - 8\mu m$, so an average area of $\pm 40\mu m^2$. How many pixels this represents in the blood film image is dependent on the magnification of the objective, and the resolution and sensor size of the camera used. The average area of an erythrocyte in pixels μ_A for the dataset is therefore pre-determined and supplied to the algorithm. Any objects larger than $2\mu_A$ are assumed to be poorly segmented cells or leukocytes, so removed from the mask. Objects smaller than $\frac{1}{2}\mu_A$ are assumed to be artefacts or over-segmented cells, so also removed. Furthermore, objects on the borders of the image are cells that are only partially in view, which makes them unsuitable for determining parasitemia, so these are also removed. The resultant mask is transferred back onto the original image, and the erythrocyte objects are cropped out.

Performance was evaluated on each of the images in the Loddo and AiDx datasets. As described above, the algorithm has severable tunable parameters and options. The optimal

values for these were determined for both of the datasets. Parameters were not tuned for each image in the dataset individually. This might have improved performance, but it would also have negated the extent to which this method could be called ‘automated’, thus undermining the benefits.

Parameters of the preprocessing steps were decided experimentally such that they resulted in a smooth and bimodal histogram. The number of dilations and erosions needed to enclose the holes inside objects, and the choice to fill them, was also decided experimentally such that the effectiveness of the watershedding algorithm was optimised. Average area of erythrocytes was measured in one image in both datasets. The value of each parameter is given in table 3-1.

Table 3-1: Table of tunable parameters in the algorithm, and the values that were used for segmenting both datasets.

Parameter	Description	Value for Loddo dataset	Value for AiDx dataset
s_H	Size of adaptive histogram region used in CLAHE (in pixels)	200	300
β	Maximum slope of intensity transfer function used in CLAHE	3	3
s_M	Size of median filter (in pixels)	4	8
ν	Number of dilations and erosions	5	0
Fill holes?	Inclusion of hole filling operation, [Y/N]	Y	N
μ_A	Average area of erythrocyte (in pixels)	13,000	20,000

3-3 U-Net based segmentation

To improve the segmentation of the AiDx dataset, a second segmentation method was implemented. This method was based on a CNN architecture called U-Net [79], which was developed specifically for the task of segmenting biomedical images. It can be trained to produce a binary or multi-class segmentation mask and has been shown to work well on various segmentation problems, even with limited training data [80].

3-3-1 Architecture

The U-Net architecture consists of a contracting path, which has a classic CNN structure as described in section 2-2, and supplements this with a symmetric expanding path, making the network U-shaped. During the contraction, the spatial information is reduced while feature information is increased. The contracting path consists of 10 convolutional layers, interspersed with max pooling layers after every second convolutional layer. The convolutional layers all have kernel size 3×3 and use Rectified Linear Unit (ReLU) activation functions. The pooling layers all have window size 2×2 , so the size of the input is halved in each of them.

The expanding path is nearly symmetrical to the contracting path, consisting of another 10 convolutional layers, but feature layers are up-sampled instead of down-sampled. The up-sampled feature layers are concatenated with the corresponding feature maps from the

contracting path, which ensures that the features that are learned while contracting the image will be used to create a spatially accurate reconstruction. These combined feature maps are the input to the first of two successive convolutional layers, after which another up-sampling operations follows.

To create the final output, a convolution with 1 kernel of size 1×1 and a sigmoid activation function is applied after the last convolutional layer in the expanding path. This results in a grayscale segmentation map of the input image.

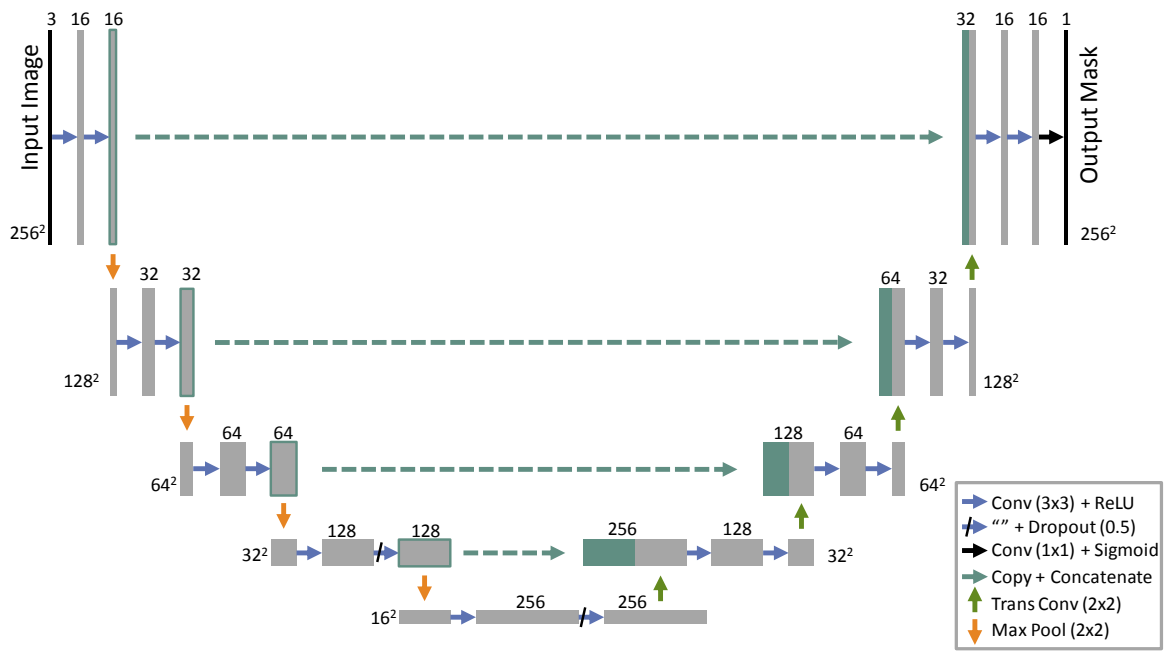


Figure 3-6: Full network architecture used for creating a segmentation map for a 256×256 RGB input image. The arrows denote convolutional (conv) and sampling operations, and the blocks denote the output of each operation. The size (width \times height) is written next to the levels, and the number of channels or depth is written above each block.

A version of this network was implemented in Python using the Keras neural network library with TensorFlow as backend [81, 82]. Some modifications and improvements to the original architecture proposed by Ronneberger et al. were made for our purpose, namely;

- Their proposed architecture was built to segment $572 \times 572 \times 1$ images, here, the number of input channels is extended to three (RGB), and to speed up learning and predictions, the input images are scaled to 256×256 .
- The number of kernels used in the convolutional layers is greatly reduced. The original architecture contained 64 kernels in the first convolutional layer and doubles after every pooling layer, here we start with 16 kernels in the first convolutional layer. This reduces the number of learnable parameters from 31,030,593 to 1,941,105, making the model much more light-weight.

- To ensure that robust features are learned and prevent over-fitting the training data, dropout with probability of 0.5 is used in two convolutions in the lowest layers of the contracting path, meaning that some of the activations are randomly set to zero in each iteration. This prevents complex co-adaptations, where high weights are assigned to features that are only useful in the context of several other specific features [83].
- In the original network, in the expanding path, the feature maps are up-sampled with a window size of 2×2 with a nearest neighbour method, after which a convolution with kernel size 2×2 is applied. We propose the use of transposed convolutions (see figure 2-5) in the expanding path instead, in which the correct weights for up-sampling are learned directly.
- Padding was added to ensure obtain an output segmentation mask the same size as the input segmentation mask. This was not the case in the original architecture, where segmentation maps of size 388×388 , containing only the middle region of the input image, were predicted. For the segmentation of full images, a tiling strategy with overlap was proposed, where the missing context in the border regions is extrapolated by mirroring the input image. This was found to add unnecessary complexity, and did not result in smooth borders between tiles. Here, padding is used to predict full segmentation maps instead, and the image is tiled with overlapping border regions, on which predictions are made twice to correct for any mistakes the padding produced.

The full architecture used is depicted in figure 3-6.

3-3-2 Training data

Training data is needed to train the network. In order to reduce computational complexity in the training, the network was not trained to segment full images from the AiDx set, but rather, square sections of the images. This also increases the amount of training objects available. The images were first down-scaled to 1024×768 , and 30 random squares with limited overlap were cropped from two of the images. Binary segmentation masks were manually drawn for these.

In order to increase the amount of training samples, data augmentation is used, which is needed to teach the network the desired invariance and robustness properties. At each training step, a new batch of training images is created by randomly applying some transformations to one of the original training images and their corresponding masks. A combination of the following transformations was used:

- **Flips** Images are randomly flipped horizontally and/or vertically.
- **Zooms** Images are randomly zoomed up to 105 %.
- **Shifts and rotations** Images are shifted horizontally and/or vertically, with a maximum of 5 % of the image size. They are also randomly rotated with a maximum of 10

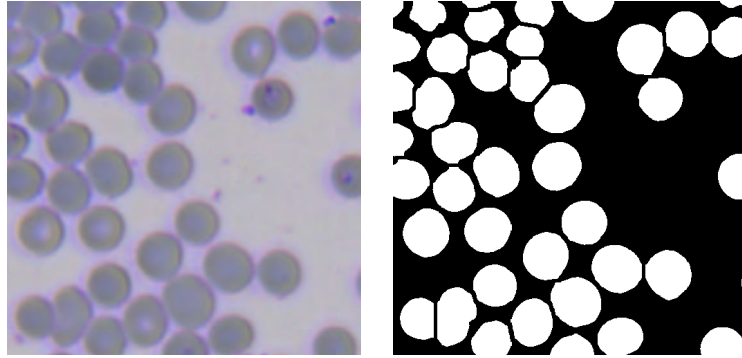


Figure 3-7: Left: 256×256 tile cropped out of an image in the AiDx dataset. Right: Hand-drawn binary mask used for training.

degrees. Both operations create ‘empty’ pixels, which are filled with the value of the nearest non-empty pixel.

- **Elastic deformations** Small random elastic deformations are applied, as proposed in [84]. This is implemented by first creating a displacement field, which defines a direction and magnitude to move each pixel in an image, and then using bilinear interpolation to apply these fields to the images. In order to create the field, first, a matrix the size of the image (256×256 in this application) is filled with values random selected from a uniform distribution $[-1, 1]$. This matrix is smoothed with a Gaussian filter, i.e. each pixel is convolved horizontally and then vertically with a vector kernel containing sampled values of a Gaussian distribution;

$$\frac{1}{\sqrt{(2\pi\sigma^2)}} e^{-\frac{x^2}{2\sigma^2}}$$

Here, $\sigma = 10$ was chosen as an appropriate standard deviation. The filter was truncated at $4\sigma + 1$, so that most of the continuous distribution area (96 %) is within the discrete kernel. After smoothing, the displacement field is multiplied with a scaling factor ϕ to achieve an appropriate distortion size. This scaling factor was set at $\phi = 150$, which resulted in images that were visibly different from the input, but still had naturally shaped cells.

Figure 3-8 shows three samples from the augmented training data set, which were all created by applying a random combination of the described transformations to the same image.

3-3-3 Training and testing strategy

The network is trained through back propagation as described in section 2-2, with Kaiming initialisation for the weights. The binary cross entropy (equation 2-7) is used as the loss function. All thirty image-mask pairs in the training set were used for training. New training data was generated in real time, by applying data augmentation to each of the samples in the training set before every epoch, meaning the (exact) same data was never passed through the network twice, but instead, thirty new augmented data samples are used every time. The

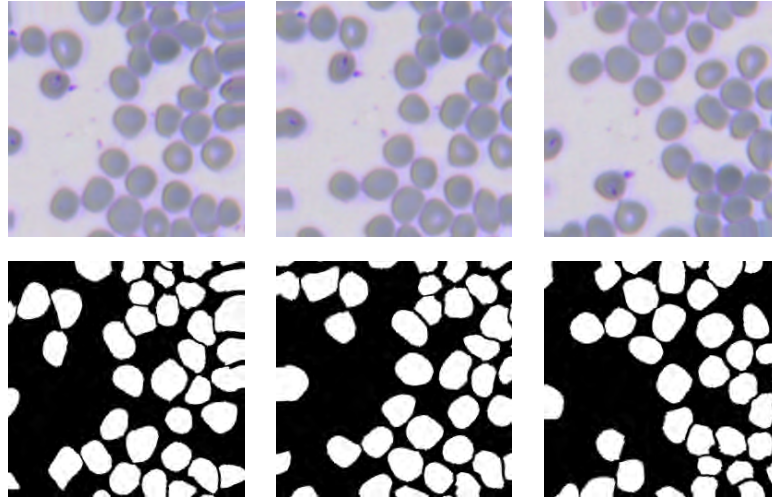


Figure 3-8: Three different augmented training samples and their corresponding masks, created by applying three sets of transformations to the image and mask in figure 3-7.

thirty samples were evenly split into batches of 10, so one epoch consisted of 3 iterations. During training, loss and accuracy are monitored.

In order to use the trained U-Net to segment full images, the following strategy is used;

1. Full images are scaled down to 768×1024 .
2. The images are divided into square 256×256 regions and cropped. In order to ensure smooth borders, there is overlap between the tiles in horizontal and vertical direction; 5×4 squares are cropped out.
3. Trained U-Net is used to predict a segmentation mask on each of the images. Due to the sigmoid activation in the final layer, the segmentation mask is grayscale (pixel intensities $\in [0, 255]$). It is converted to a binary image by thresholding at 70 %. This results in a mask with white erythrocytes on a black background.
4. The masks are stitched back together into a full size image. In the overlapping regions, the pixel value is taken as the maximum value of the two overlapping images, so if the region is classified as erythrocyte in either image, it will be in the final segmentation mask. This was done because the zero padding used sometimes resulted in under-predicting cell instances in the borders of the segmentation masks.
5. The final mask is scaled back up to full image size (4912×3684) and used to crop cells out of the original image. As was the case in the thresholding based method, any objects touching the borders or outside the size range $[\frac{1}{2}\mu_A, 2\mu_A]$ are ignored.

This segmentation method was primarily evaluated on the six images of the AiDx dataset, since it was trained specifically to segment those images. In order to see if this method generalised to other data, despite only being trained to segment the AiDx data, performance was also evaluated on the Loddo set.

3-4 Classification

In order to determine infection in segmented erythrocytes, a transfer learning strategy is used; meaning an existing CNN architecture is modified to classify the images.

The existing architecture that was used for this task is the VGG-16 network which was developed by Simonyan et al. (of the Visual Geometry Group from Oxford) [85]. This is a very large-scale network, the ‘16’ in the name refers to 16 layers with trainable weights: 13 convolutional of increasing depth and 3 fully connected, making the total number of learnable weights in the network 138 million. Training a network of this size takes tremendous computational power and is not feasible within the context of this project. However, the network has been trained exhaustively on the ImageNet database [86], which contains an enormous variety of images in many different classes, and the weights obtained are available. The image features that the convolutional network was trained to extract on these images, are potentially also suitable for our image classification problem.

3-4-1 Network architecture

The full architecture of the network is as follows: two sets of two convolutional layers are followed by three sets of three convolutional layers. The sets are interspersed with four max pooling layers, which have a window size of 2x2 and a stride of two, so the size of the channels is halved after every max pooling layer. The first set of convolutional layer contains 64 kernels, and the number of kernels doubles after every pooling layer except the last one. All kernels in the network are of size 3×3 and use a ReLU activation function.

In the published network, the convolutional layers are followed by another Max Pooling layer, and three fully connected layers. These are all omitted in the architecture used here. Instead, the final convolutional layer is followed by a Global Average Pooling layer, as proposed in [87]. This reduces the extracted $6 \times 6 \times 512$ feature map, to a one-dimensional $1 \times 1 \times 512$ feature map, by taking the average value. The result is the input to a fully connected layer with 1024 neurons which uses a ReLU activation function. During training, these neurons are dropped out with a probability of 0.5. This is followed by a fully connected layer with two neurons and a soft-max activation function, which maps the output to a two-dimensional vector containing two class probabilities; probability of belonging to class ‘infected’ p_0 , probability of belonging to class ‘uninfected’ p_1 . Obviously, since this is binary classifier, $p_1 = 1 - p_0$. The architecture of the VGG-16 network is depicted in 3-9.

This network was implemented in Python using the Keras neural network library with TensorFlow as backend [81, 82].

3-4-2 Training and testing strategy

The pre-trained weights for the convolutional layers of the VGG-16 network are downloaded from the Keras library. The weights in the two fully connected layers were initialised with Kaiming initialisation. The data is propagated forward and backward through the entire network at each iteration, but only the weights in the fully connected layers are updated. This means, the convolutional layers act as a static feature extractor, which provides the

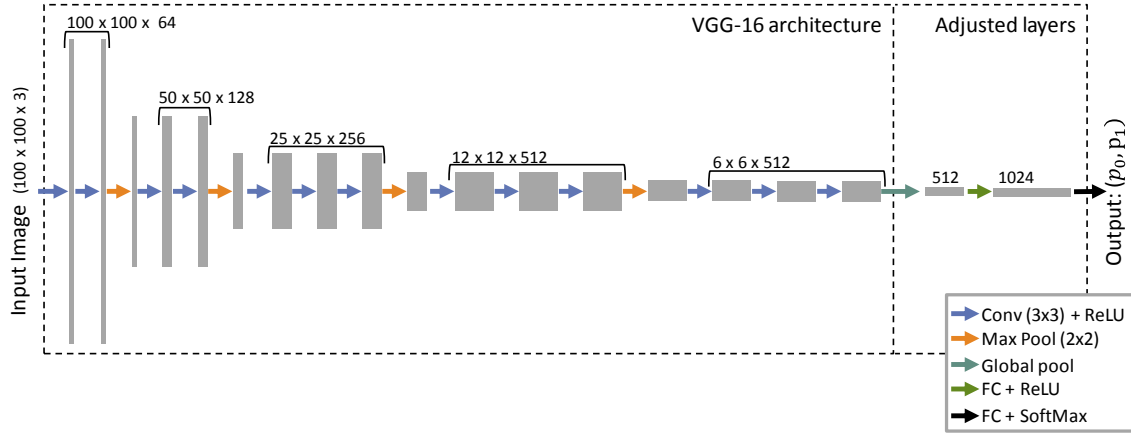


Figure 3-9: The architecture used for classification of segmented erythrocytes. The arrows denote convolutional (conv), pooling and fully connected (FC) operations, and the blocks denote the output of each operation. The output size is written above the blocks (width \times height \times depth), note that the output of the fully connected layers is vectorised so one-dimensional. The convolutional layers of the VGG-16 net and their pre-trained weights on the ImageNet database are not adjusted, only the final two fully connected layers are.

input to the simple Artificial Neural Network (ANN) which consists of only 1 hidden layer. Data from the Rajaraman dataset is used to train the network. All images are scaled to 100×100 . These are divided into training and validation data with an 80 / 20 split, so 11,023 samples per class are used for training and the remaining 2,756 are used to evaluate performance. The batch size is set at 32, and the performance of the network in terms of loss and accuracy is evaluated after every epoch (250 iterations) on the training and validation data. The binary cross-entropy (equation 2-7) is used as loss function.

In order to accelerate gradient descent in the relevant direction and dampen oscillations in the network updates, the gradient descent method as described in equation 2-11, is modified here to include ‘momentum’, meaning a weighted average of the previous updates is used to determine the next update U ;

$$U_t = \beta U_{t-1} + \alpha \frac{\partial J}{\partial w} J \quad (3-8)$$

Here, U_t refers to the update at this step and U_{t-1} is the update at the previous step. So after every iteration, the weights are now updated by $w \leftarrow w - U_t$. The scaling factor β is set at 0.9, and the learning rate is set at $\alpha = 10^{-5}$.

In addition to evaluating performance of the trained network on the validation set, it is also evaluated on the AiDx dataset, by applying it to make predictions on the erythrocytes images produced by the best of the two segmentation methods, which resulted in a dataset with 2176 objects in total. In order to validate the predictions of the network, the pre-determined ground truths were used to divide these objects into two classes. Some objects that were not erythrocytes were found in the segmentation, which were assigned to the class uninfected. It is to be expected that this limits performance, but it does provide insight of the performance of the entire diagnostic procedure when segmentation and classification are implemented back-to-back. This resulted in 202 infected cell objects and 1974 uninfected ones.

In order to improve performance specifically on the AiDx dataset, the network was trained on this image data. The same training strategy was used, except the learning rate was lowered to $\alpha = 10^{-6}$ to prevent large gradient updates away from the learned activations. The erythrocytes objects of 5 image were selected to use as training data, with non-cell objects removed to prevent learning wrong features, and the final image was reserved for testing, resulting in a training set of 1564 uninfected and 163 parasitised cells.

Training on this unbalanced dataset would be problematic, as the predictions would quickly converge to favour uninfected. In order to remedy this, data augmentation is used on the objects in the parasitised class; 9-10 augmented images were created from each sample, to achieve equal class sizes. Images were flipped, rotated at angles of 90 degrees, and random elastic deformations as described in section 3-3-2 were applied with $\sigma = 7$ and $\phi = 100$. An example of the resultant images is shown in figure 3-10. As can be seen, applying the elastic deformations causes the cell to lose their natural round shape. However, since a lot of objects in the training and validation set used here were shaped irregularly to begin with, as a result of the segmentation method used, this was not considered a problem.

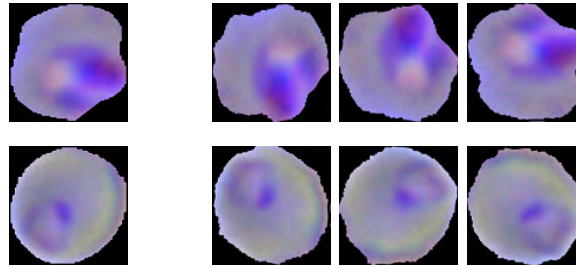


Figure 3-10: Two parasitised erythrocytes from the AiDx images resized to 100×100 , with three augmented training samples based on on each. Original, non-augmented images are shown on the left.

Chapter 4

Results

In this chapter, the results of the segmentation and classification algorithms on the different datasets are presented. Firstly in section 4-1-1, the results of the threshold based segmentation method on the Loddo and AiDx dataset are discussed. Types of errors made by the algorithm are shown, and quantified for all images separately. From this, performance measures for both datasets are calculated and compared.

In section 4-1-2, the results of the U-net based segmentation method are discussed. First, the learning results of the neural network are presented. Then, segmentation performance is presented and compared to the performance of the first segmentation method. As the U-net was specifically trained to segment AiDx data, the focus in this section is on the performance on this dataset, but performance on the Loddo set is also discussed briefly.

Finally, in section 4-2, the classification results are presented. The VGG-16 based network is first trained on the Rajaraman data, training and validation results for this are presented in section 4-2-1. Performance of this network is also evaluated on the erythrocytes that were segmented out of the AiDx images. This is done on an image-by-image basis, allowing us to predict parasitemia estimates for each image, which are compared with the ground truth.

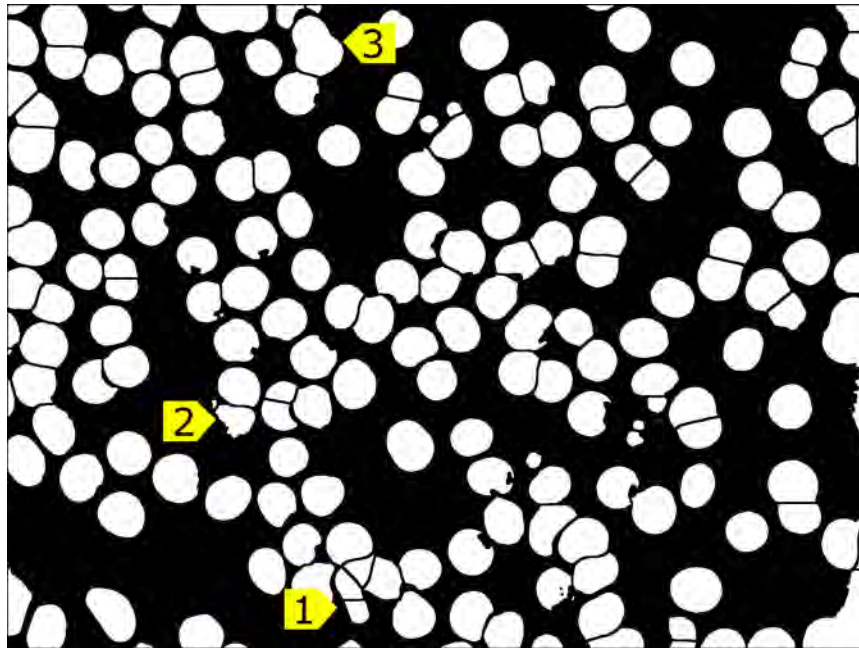
The network is then retrained on the majority of the segmented AiDx data, for which the results are presented in section 4-2-2. Performance is evaluated on the one image from the AiDx dataset which was not used for training, and compared with performance of the network before retraining.

4-1 Segmentation results

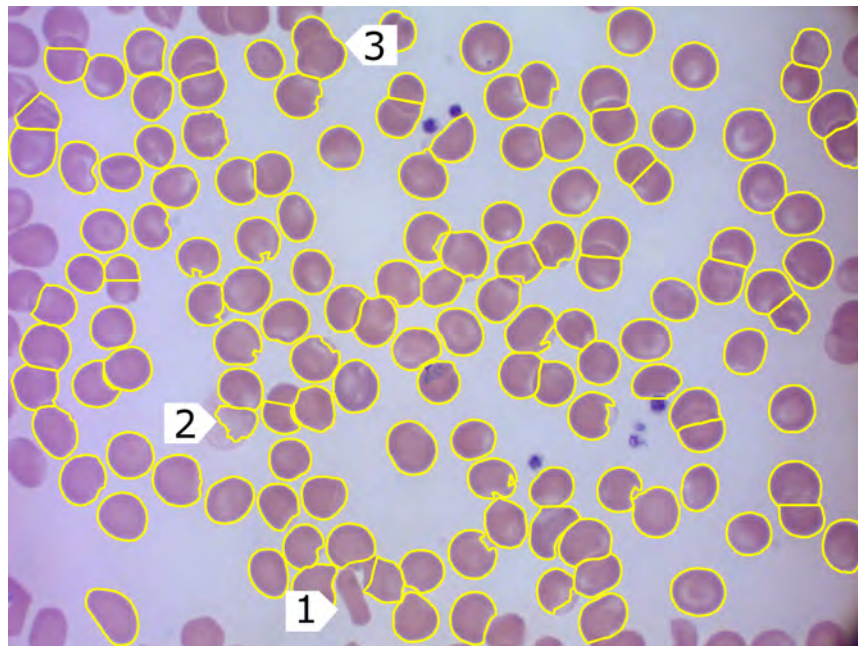
4-1-1 Segmentation results with threshold method

The algorithm described in section 3-2 was applied to each of images in the Loddo and AiDx datasets. The result for the first image in the Loddo dataset is shown in figure 4-1.

To quantify the performance of the segmentation method, results were compared with a ground truth, in which cell locations were manually appointed. For each object found, the



(a) Full binary segmentation mask, before removing objects that have touching pixels with the border and objects that are outside the size range.



(b) Original image with segmentation overlay.

Figure 4-1: Segmentation result for image no. 1 in the Loddo dataset. Three wrong results are pointed out; at **1**) over-segmentation occurred, resulting in multiple small objects, which were all removed before applying the segmentation mask, resulting in a false negative (FN), at **2**) a non-erythrocyte object was found (false positive (FP)) and at **3**), under-segmentation occurred (FN).

number of manual ground-truth points in that objects region was checked. If there was exactly one point in the region, this was counted as a true positive (TP). If there was no point in the region, this was counted as a FP. If there were two points in the region, the object consisted of multiple cells, so under-segmentation had occurred. This was counted as a FN, since one cell was missed in this case. Theoretically, this could be extended to objects containing more than two points. However, since any objects larger than two times the average cell size were deleted when applying the mask, none were found.

The watershed algorithm sometimes resulted in over-segmentation, i.e. a cell was split into two or more objects. When this occurred, often, one or more of the resultant objects were smaller than half the average cell size so deleted. Only objects that covered at least half an average cell area were kept, and for these, the standard rule was applied: an object that contained the ground truth point was counted as TP and any other objects were counted as FP.

Three examples of errors in segmentation are highlighted in figure 4-1.

Three performance measures were calculated for each of the images. The true positive rate (TPR) or recall / sensitivity is a measure of the proportion of ground truth erythrocytes that are correctly identified, calculated as;

$$TPR = \frac{TP}{TP + FN} \quad (4-1)$$

The positive predictive value (PPV) or precision, is the likelihood that a positive call is indeed an erythrocyte, calculated as;

$$PPV = \frac{TP}{TN + FP} \quad (4-2)$$

The F_1 -score is a measure of the accuracy of the test, also known as the precision-recall score since it is the harmonic mean of the two, calculated as;

$$F_1 = 2 \times \frac{PPV \times TPR}{PPV + TPR} \quad (4-3)$$

The results for the Loddo dataset are given in table 4-1, the results for the AiDx dataset are given in table 4-2. On both datasets, the average PPV is high; ± 0.98 for both. This means that on average only 2% of detected objects are not erythrocytes. This is important when segmentation and classification are implemented back to back; the classifier is not trained to detect non-erythrocyte objects, and would not produce useful results when presented with them. The average sensitivity for the Loddo dataset is also high (0.976), but on the AidX dataset the sensitivity is significantly lower (0.895). This means that about 10 % of cells were not detected as objects with this algorithm.

Table 4-1: Test results for threshold based segmentation algorithm on the Loddo dataset. For each individual image, the number of objects that were TP, FP and FN are given, and the performance measures were calculated from those.

Image no.	TP	FP	FN	TPR	PPV	F_1 -score
1	137	1	2	0.986	0.993	0.989
2	157	4	9	0.946	0.975	0.960
3	146	10	10	0.936	0.936	0.936
4	165	6	6	0.965	0.965	0.965
5	146	11	10	0.936	0.930	0.933
6	108	0	0	1.000	1.000	1.000
7	153	0	5	0.968	1.000	0.984
8	145	0	0	1.000	1.000	1.000
9	135	0	0	1.000	1.000	1.000
10	114	0	1	0.991	1.000	0.996
11	133	0	2	0.985	1.000	0.992
12	168	0	0	1.000	1.000	1.000
13	151	0	0	1.000	1.000	1.000
14	132	1	5	0.964	0.992	0.978
15	128	0	0	1.000	1.000	1.000
16	163	1	2	0.988	0.994	0.991
17	187	5	10	0.949	0.974	0.961
18	170	6	3	0.982	0.966	0.974
Average	147	2.5	3.6	0.976	0.985	0.981

Table 4-2: Test results for threshold based segmentation algorithm on the AidX dataset. For each individual image, the number of objects that were TP, FP and FN are given, and the performance measures were calculated from those.

Image no.	TP	FP	FN	TPR	PPV	F_1 -score
1	224	7	23	0.907	0.970	0.937
2	350	15	47	0.882	0.959	0.919
3	280	6	40	0.875	0.979	0.924
4	314	6	41	0.885	0.981	0.930
5	371	4	48	0.885	0.989	0.935
6	412	6	28	0.936	0.986	0.960
Average	325	7.3	37.8	0.895	0.977	0.934

When inspecting the segmentation masks of the AiDx images, it becomes clear what causes the lower sensitivity. Larger groups of closely clustered cells could not adequately be separated with watershedding, resulting in objects with size $> 2\mu_A$, which were omitted when the segmentation mask was transferred. An example of this happening is shown in figure 4-2.

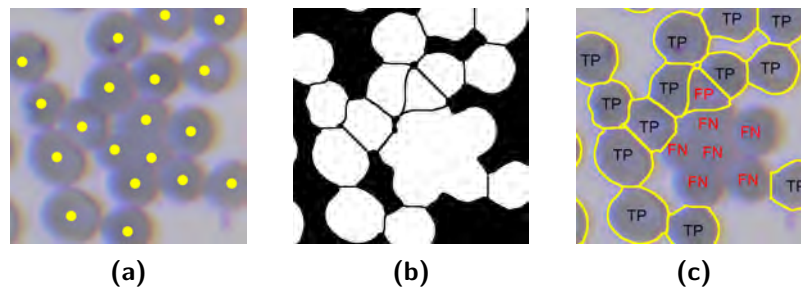


Figure 4-2: Error of threshold segmentation algorithm in image no. 5 from the AiDx dataset; **a)** is a section of the original image, with ground truth points overlaid, **b)** is the corresponding binary segmentation mask and **c)** shows the result of applying the segmentation mask to the original image, with hits and misses.

4-1-2 Segmentation results with U-Net method

Training results

Accuracy and loss were calculated at the end of every iteration on the batch of augmented training data that was used. Accuracy is the fraction of pixels in predicted masks that exactly match their ground truth and loss is the binary cross-entropy. The average of these measures over the three iterations was calculated at the end of each epoch. Training was stopped when significant improvements were no longer made, after 50 epochs. The training progress is depicted in figure 4-3. At the end of training, the loss was calculated on the original data, which was not used in training without augmentation. On this data, the loss was equal to 0.110 and the accuracy was 95.3 %.

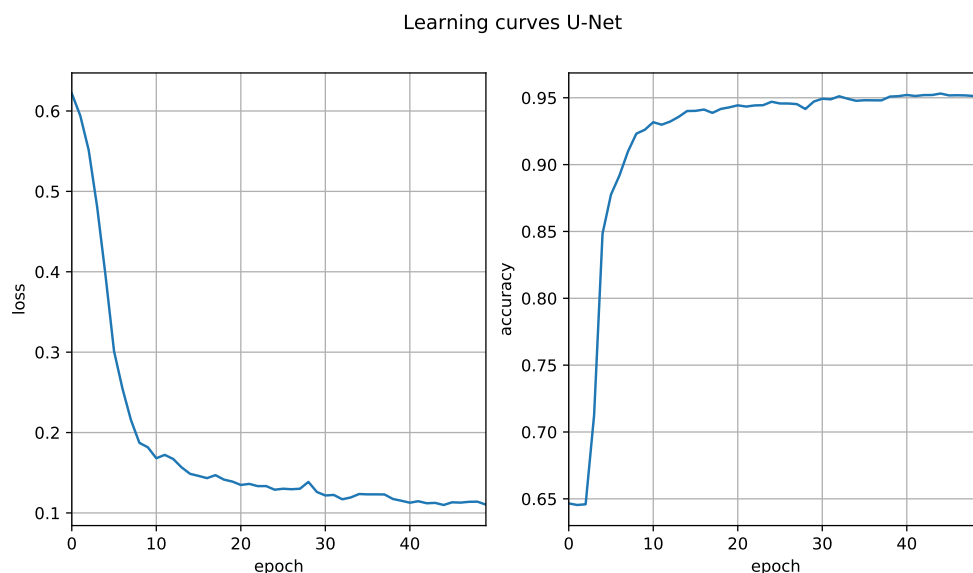


Figure 4-3: Progression of loss and accuracy of the U-Net during training. Both performance metrics were calculated at the end of each iteration on the batch of augmented training data that was used at that step. The values in this graph are the average over the three batches in the epoch.

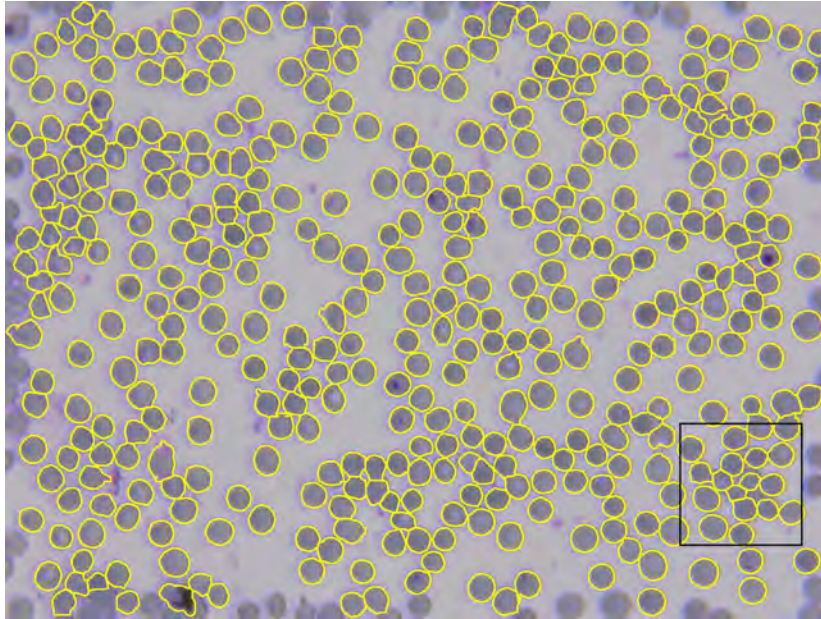


Figure 4-4: Image no.5 of the AiDx dataset, with segmentation overlay produced by U-Net algorithm. To allow for comparison between the two segmentation methods, the square indicates the area for which the results of the threshold algorithm were depicted in figure 4-2.

Segmentation results

After training, the network was used to create segmentation masks for each of the images in the AiDx dataset. The result for one of the images is shown in figure 4-4.

The performance of the network was evaluated in the same way as described in section 4-1; segmented objects were compared with ground truth points, and the amount of TP, true negative (TN) and FP objects was determined. The results are given in table 4-3.

The first two images had been used to extract training samples from, although differently

Table 4-3: Test results AiDx dataset U-Net based segmentation

Image no.	TP	FP	FN	TPR	PPV	F_1 -score
1	247	0	0	1.000	1.000	1.000
2	396	1	1	0.997	0.997	0.997
3	319	2	1	0.997	0.994	0.995
4	355	0	0	1.000	1.000	1.000
5	416	1	3	0.993	0.998	0.995
6	440	0	0	1.000	1.000	1.000
Average	362	0.7	0.8	0.998	0.998	0.998

cropped regions were used there than in the evaluation. The other four were completely unseen. If the network had over-fitted on the training set, it would be expected that performance on the first two images would be better. Since this is not the case, it can be concluded that overfitting did not occur.

The average TPR, PPV and F_1 -score of this method are all 0.998, which indicates signif-

icantly better performance than the the threshold based algorithm. This can be observed by comparing the area indicated by a square in figure 4-4, with the result of the threshold method for the same area, which was depicted in 4-2.

Performance on the Loddo dataset was also evaluated. Average sensitivity, precision and F1-score over eighteen images were 0.885, 0.997 and 0.933 respectively. The lower sensitivity was to be expected, since the U-Net was only trained on the AiDx image data, so it only recognises erythrocyte objects that look similar to those. When inspecting the results, it became clear that the classifier especially does not recognise central pallor as being part of the cell object, resulting in over-segmentation. This makes sense, because central pallor was minimally visible in the AiDx images.

4-2 Classification results

4-2-1 Results for network trained on Rajaraman data

Training results

The network was trained on the Rajaraman data, the learning curves are depicted in 4-5. Loss decreases rapidly during training while accuracy increases, on both the training and validation data, meaning the network is learning the correct predictions. At the end of training, the training and validation curves show a good fit; indicating that the features of the validation set are represented well by the training set. At the beginning of training, loss on the validation set is consistently better than loss on the training set, which makes sense, since loss on the training set is taken as the average over the entire epoch, and loss on the validation set is evaluated with the network parameters that are learned at the end of the epoch. Learning was stopped after 100 epochs, since the loss on the validation set was no longer decreasing significantly.

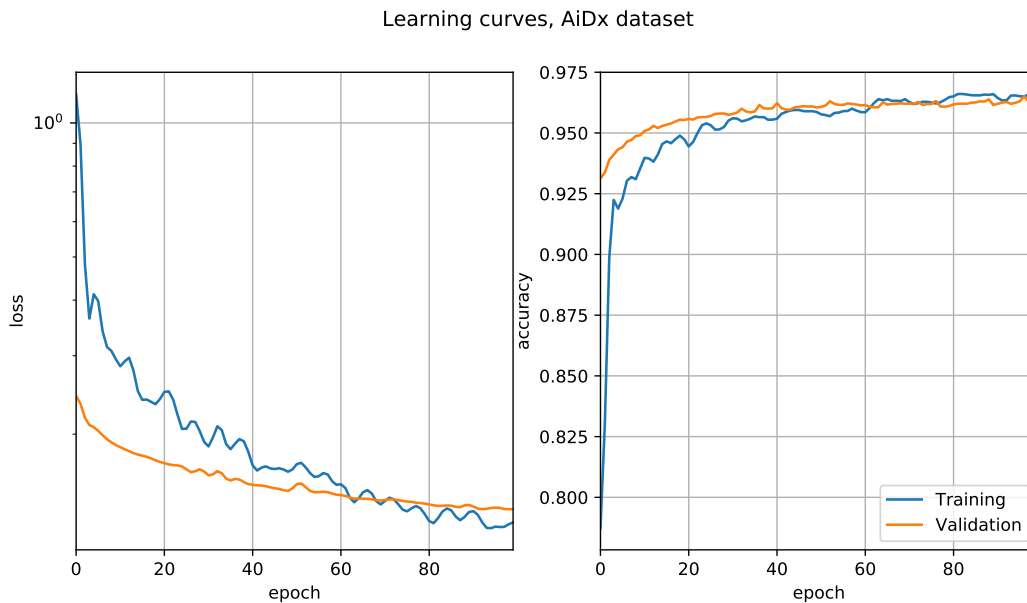


Figure 4-5: Progression of loss and accuracy during training of the classification network on the Rajaraman data. The loss is plotted on a logarithmic scale.

Classification results on validation sets

After training, the saved model was used to make final predictions on the Rajaraman validation set. Performance was also evaluated of the segmented erythrocytes from the AiDx dataset, as produced by the U-Net based method, which had the best segmentation performance. The predicted class probabilities were used to assign classes to each instance; when $p_0 > p_1$, the objects were assigned to class 0 (infected) and when $p_0 < p_1$, the objects were assigned to class 1 (uninfected).

The results on both datasets are shown in the form of a confusion matrix in 4-6. The labels predicted by the network are given in the columns, while the rows represent the ground truth

class instances, the totals of which are given to the left of the column. In this way, the upper left and lower right corners are the absolute amount of TPs and TNs respectively, and the lower left and upper right corners are the FPs and FNs respectively.

		Predicted labels		
		Infected	Uninfected	
True labels	Infected	2712	45	Total: 2757
	Uninfected	146	2611	Total: 2757

(a) Rajaraman validation set

		Predicted labels		
		Infected	Uninfected	
True labels	Infected	157	45	Total: 202
	Uninfected	220	1754	Total: 1974

(b) AiDx validation set

Figure 4-6: Confusion matrices for both validation sets, showing the total number of correct and incorrect predictions made on each set.

The values in figure 4-6 were used to calculate performance measures for the classifier. The sensitivity (TPR), precision (PPV) and F_1 scores were calculated as described in eqs. (4-1) to (4-3). Additionally, accuracy and specificity were calculated. Accuracy is the total rate of correct predictions, given by;

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \quad (4-4)$$

Specificity, also known as true negative rate (TNR) is calculated by;

$$TNR = \frac{TN}{TN + FP} \quad (4-5)$$

The performance measures are summarised in table 4-4. It is clear that while good performance was achieved on the Rajaraman validation data, performance on the AiDx validation data is not as good, especially in terms of sensitivity and precision. It is plausible that the parasitic features in the AiDx erythrocyte images, which were acquired with a different set-up (different camera, illumination, magnification etc.) were not well represented by the Rajaraman dataset that was used to train the network, leading to the diminished performance.

Table 4-4: Performance measures of the trained network on the Rajaraman validation set and the segmented erythrocytes from the AiDx dataset.

	Accuracy	TPR	TNR	PPV	F_1 -score
Rajaraman validation set	0.965	0.984	0.947	0.949	0.966
AiDx validation set	0.905	0.795	0.915	0.477	0.636

The values and performance measures given in figure 4-6 and table 4-4, are calculated for the situation where objects are assigned to the class with the greatest predicted probability. In

other words, since this is a binary classification problem, when $p_0 > 0.5$ objects are assigned to class 0, and when $p_0 < 0.5$, objects are assigned to class 1. It is possible to choose a different value than 0.5 for the classification threshold, to satisfy different performance criteria. The range of possible operating points is depicted by plotting the Receiver Operating Characteristic (ROC), which is determined by choosing different thresholds $\tau \in [0, 1]$, and computing sensitivity and specificity at these thresholds. The ROC on both validation sets is shown in 4-7.

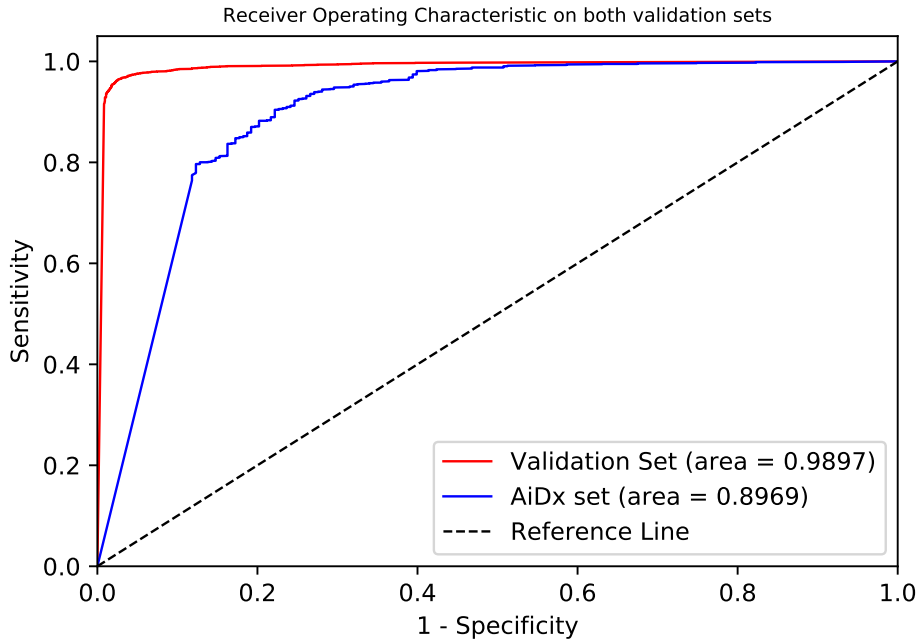


Figure 4-7: Receiver Operating Characteristic curves for both validation sets. Sensitivity and specificity are plotted for different thresholds, starting at $\tau = 1$ on the left (all objects are assigned to class uninfected) and ending at $\tau = 0$ on the right (all objects are assigned to class parasitised) Area under the curve is given in lower right corner. The reference line depicts a theoretical 'random guessing' classifier (i.e. on that is right 50 % of the time at $\tau = 0.5$).

For an ideal classifier, sensitivity and specificity would be equal to 1 at any threshold $\tau \in (0, 1)$, and the area under the ROC would be equal to one. This means that the larger the area under the curve is for a given validation set, the more closely the classifier approximates an ideal classifier for that dataset. In general, when choosing a higher threshold, specificity is improved at the cost of sensitivity, and vice versa. Any operating points that lie above the line are not achievable by the classifier, so for instance, if we wish to have at least 90 % sensitivity on the AiDx data, we can at most be 75 % specific.

To gain insight into the performance of both the U-Net segmentation algorithm and the VGG-16 based classifier when implementing both back to back, to go automatically from raw image to parasiteamia estimate, the performance of the classifier was also evaluated for each of the images in the AiDx dataset separately. The number of total infected objects found in each image was divided by the number of total segmented erythrocyte objects found in that image to obtain a parasiteamia estimate. These were compared to the ground truth, full

results are given in table 4-5. It is clear that the low precision leads to over-estimation of the parasiteamia in these images.

Table 4-5: Ground truth cell and infection counts, compared with estimates for both by the U-Net based segmentation method and the VGG-16 based classifier respectively. Ground truth and estimated parasiteamia levels are given, and compared in the final column.

Image no.	Ground truth values			Values estimated by algorithm			% Of true parasiteamia
	Cells	Infected	Parasiteamia	Cells	Infected	Parasiteamia	
1	247	33	0.1336	247	47	0.1903	142
2	396	39	0.0985	396	56	0.1414	144
3	320	31	0.0969	321	58	0.1807	187
4	355	25	0.0704	355	45	0.1268	180
5	418	35	0.0837	417	106	0.2542	304
6	440	38	0.0864	440	65	0.1477	171

The estimated erythrocyte and parasite locations in the AiDx images were visualised by plotting them onto the original images. Results for one image are shown in figure 4-8.

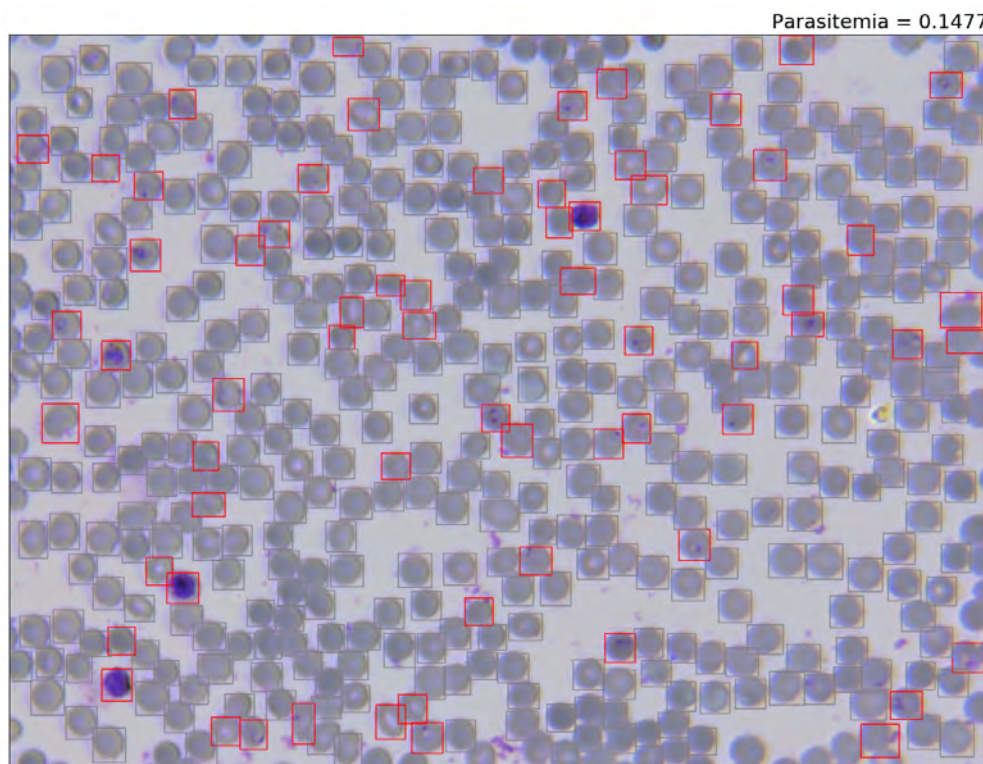


Figure 4-8: Predicted locations of healthy erythrocytes, indicated with a gray bounding box, and parasitised erythrocytes, indicated by a red bounding box, in image no. 6 from the AiDx dataset.

4-2-2 Results for network trained on AiDx data

Training results

The model weights were initialised at the weights found in the previous section. Data of the image no. 6 was reserved for validation, while the rest of the (augmented) training data was used to estimate the correct weights on the AiDx dataset. The learning curves are shown in figure 4-9. It is clear, that the data we reserved for validation was better represented by the original model than the data in the training set was, as at the start of training, the loss is lower on the validation set and the accuracy higher. However, as the network starts to learn, the two curves converge. It was observed that after ± 60 epochs, the loss on the validation set remained static and later even started to go up, while the loss on the training set keeps decreasing, indicating that the network starts to over-fit the training data. Based on this, learning was halted after 60 epochs and the model weights that were saved at epoch 60, were kept and used to evaluate performance on the validation data.

Compared with figure 4-5, it can be observed that loss and accuracy updates are more stochastic, even though a smaller learning rate was used. This is explained by the fact that both the training and validation sets were far smaller than the ones used previously, so the updates are averaged over less samples, causing greater variation.

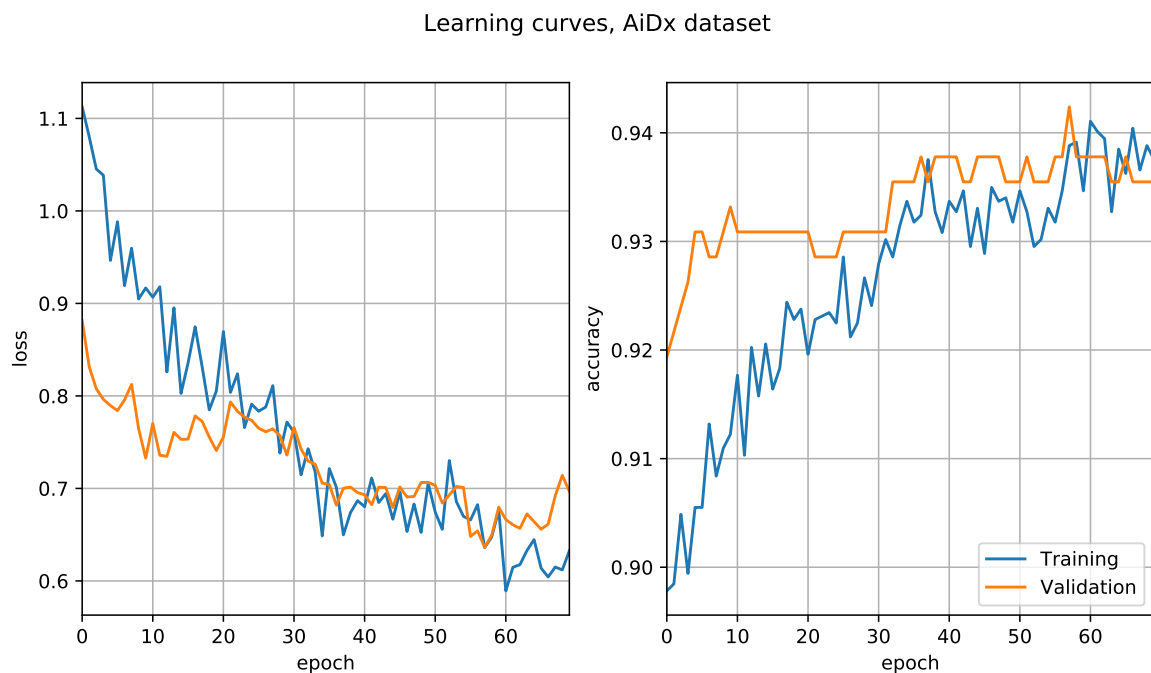


Figure 4-9: Progression of loss and accuracy during training of the classification network on the AiDx data.

Classification results on validation set

The retrained network was used to predict on the validation data set (segmented erythrocytes objects of AiDx image no. 6), the confusion matrices for this data before and after retraining are shown in figure 4-10. After retraining, the number of FPs declined, while the number of

FNs remained the same. In fact, when comparing results between both network predictions, the exact same cells were still marked as FNs.

The performance measures calculated from this are given in table 4-6. Specificity increased,

		Predicted labels		
		Infected	Uninfected	
True labels	Infected	30	8	Total: 38
	Uninfected	35	367	Total: 402

(a) Before retraining

		Predicted labels		
		Infected	Uninfected	
True labels	Infected	30	8	Total: 38
	Uninfected	20	382	Total: 402

(b) After retraining

Figure 4-10: Confusion matrices showing the number of correct and incorrect predictions on erythrocytes from AiDx image no. 6, before and after retraining the network.

as did precision, but sensitivity remained the same; the network learned to better recognise uninfected cells; but did not perform better in recognising parasitised cells. This is not unexpected; the number of objects in the uninfected class available for re-training was a lot bigger than the number of training objects in the parasitised class, making it likely that not all parasite features present in the validation set were represented in the training set. This in turn makes it impossible for the network to learn to distinguish these objects as parasites.

Table 4-6: Performance measures on erythrocytes from AiDx image no. 6, before and after retraining the network.

	Accuracy	TPR	TNR	PPV	F1-score
Before retraining	0.902	0.789	0.913	0.462	0.626
After retraining	0.934	0.789	0.948	0.588	0.689

In 4-7, the updated parasiteamia estimate is given. A clear improvement on the previous estimate is noted.

Table 4-7: Ground truth parasiteamia for AiDx image no. 6, compared with prediction by network before and after retraining

Source	Cell count	Infected cells	Parsatimea	% Of true parasiteamia
Ground truth values	440	38	0.0864	100
Estimated by old network	440	65	0.1477	171
Estimated by retrained network	440	50	0.1136	132

To gain some visual insight into which objects causes the algorithm to make wrong predictions, the predictions are plotted onto the original image, with their ground truth score. The result is shown in 4-11

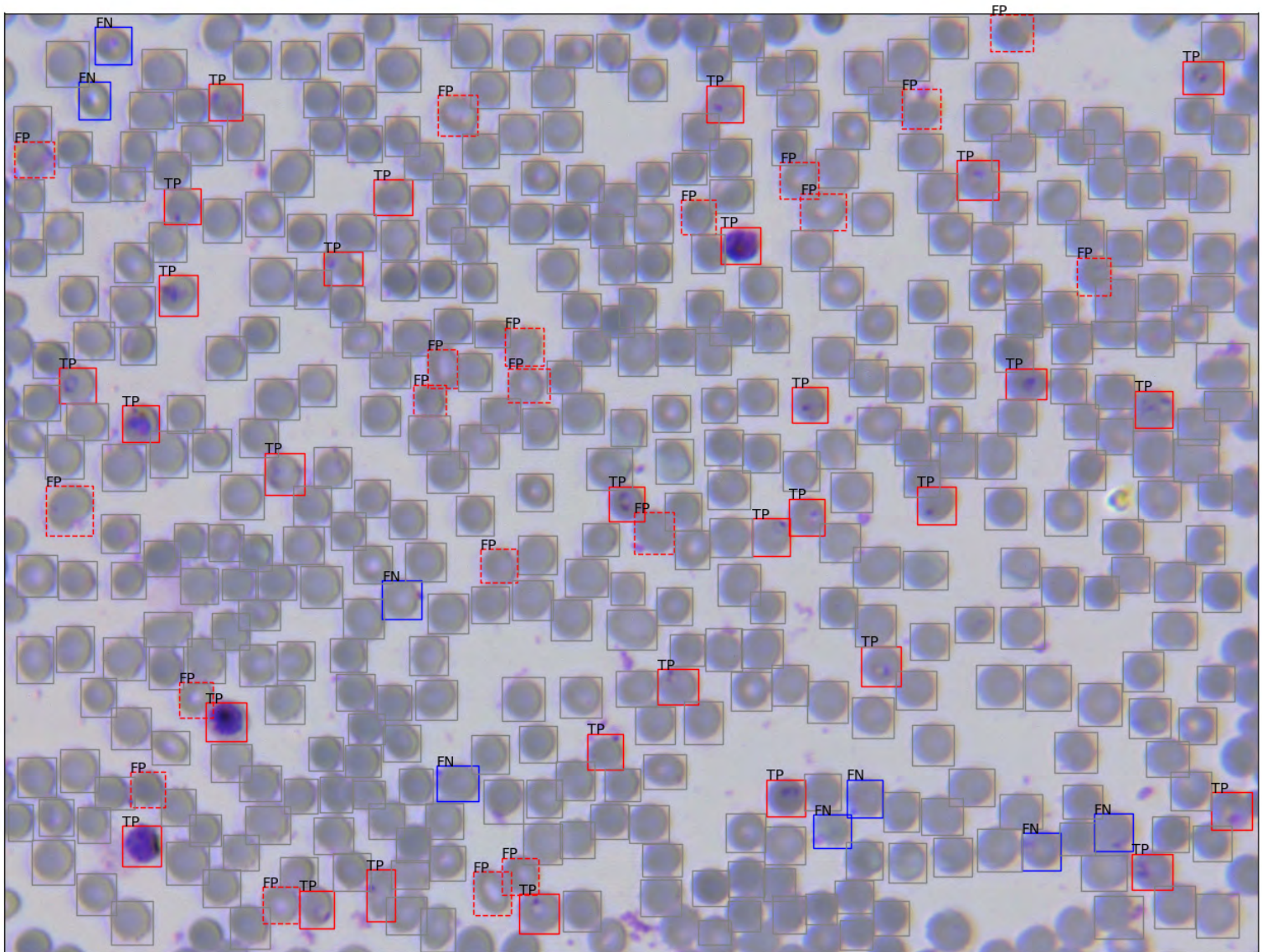


Figure 4-11: Locations of cells as estimated by the U-Net algorithm, combined with locations of parasites as estimated by the retrained VGG-16 based classifier, in image no. 6 from the AiDx dataset. Predicted parasite locations are indicated by a red bounding box; TP cells are indicated with a solid line while FP cells are indicated with a dotted line. FN cells are indicated by a blue bounding box and TN cells indicated with a gray bounding box. To compare performance, see 4-8, in which the prediction by the network on this same image before retraining is depicted.

Chapter 5

Discussion

In this chapter, the results will be interpreted and compared with the theoretical performance of a human expert, as defined by the World Health Organization (WHO). Limitations to the methods used are discussed. Some recommendations for modifications to the method used that could improve performance are made here, as well as further steps that could be taken to validate the results. More general recommendations for future work on this research topic are given in the final paragraph of this chapter.

The simple threshold segmentation method proposed exhibited good performance (average F_1 -score = 0.981) on the Loddo data set, which contained high magnification blood smears which were generally of good quality, although overlapping cells were present, which were mostly successfully separated through watershedding. Performance on the AiDx image data, which was taken at lower magnification, was not as good; average positive predictive value (PPV) was still sufficient (0.977), but the sensitivity was lower (0.895), combining to an F_1 -score of 0.934. The high PPV was achieved by removing any objects from the segmentation masks that were much smaller or larger than the average cell area, which in large part took care of removing over- and undersegmented objects.

When continuing to classify the segmented objects produced by this algorithm, the low sensitivity for cell segmentation would not necessarily lead to bad parasitemia estimates, assuming that an equal rate of uninfected and parasitised cells were missed. However, the total number of cells that are classified per image would be lower than the total number of cells in that image, thus effectively limiting the Field of View (FOV), and needing to evaluate more images before a diagnosis can be reached.

The U-Net based segmentation method showed a big improvement in performance when evaluating on the AiDx data, with an average sensitivity, PPV and F_1 -score of 0.998. This means that nearly all objects that were found were indeed erythrocytes, and nearly no erythrocytes were not found. Comparing with segmentation methods that were described in table 2-1, this segmentation method outperforms the one that was proposed by Rajaraman et al. (PPV 0.941), and the one proposed by Delgado-Ortet et al. (accuracy 0.937). Though lower performance was achieved on the Loddo dataset, it is believed that this network can be trained with different data to produce good results on other datasets as well. An interesting research

question to explore in future work, would be to see if a network of this type can be made more robust to changes in input data, and be trained to produce good results on both datasets without retraining in between. Though U-Net-type networks have been used on various datasets in literature [80], no research was found in which this option was explored.

The performance measures of both segmentation methods were expressed in terms of correct objects found, which does not provide insight into how well the cell border was detected. Since cells were located closely together, any very wrong cell borders would still lead to lower scores on these measures, so they were deemed sufficient to validate performance for our purposes. Some objects were however found that only covered part of the cell, or included staining artefacts that were touching the cell. These were counted as TP, so this was not reflected in the performance measures.

It could be said, that the performance of these methods in terms of erythrocyte localisation has been validated, but segmentation performance was not quantitatively demonstrated. A good metric to do so, would be to calculate the Intersection over Union (IoU), in which the overlap of the predicted erythrocyte object with its ground truth (intersection), is divided by the total area encompassed by both the ground truth and the predicted object (union)[88]. However, calculating the IoU would require knowing the exact ground truth location and shape of each erythrocyte in each image, which were not available and would have to be created manually. The validation method used here was deemed more efficient, and would also be far less labour intensive to extend to new validation data.

We can compare the results of the classification method with the performance requirements the WHO sets for human microscopists, which were given in table 2-2. Ideally, we would like our algorithm to perform as well as a ‘level 1’ expert microscopist on the AiDx data; which means achieving a sensitivity $\geq 90\%$, and a parasiteamia estimate within 25% of the true parasiteamia at least 50% of the time.

This sensitivity was more than achieved by the classifier on the Rajaraman validation data, on which an average accuracy, sensitivity and specificity of 96.5 %, 98.4 % and 94.7 % were achieved, outperforming all of the methods summarised in table 2-1 in terms of sensitivity, and ranking high among them in total accuracy as well. Combined with the high PPV that was achieved with this classifier, it is likely that good parasiteamia estimates would be made if this classification was done on patient level.

However, on the AiDx image data, average sensitivity was 79.5 % over 6 images, which narrowly puts the algorithm at the same level as a ‘level 2’ human microscopist. This is still deemed acceptable for certification by the WHO.

Sensitivity was not improved when retraining on the obtained segmented erythrocyte data, however, this can be attributed to the low volume of training data available. Data augmentation was applied to the images in the parasitised class, but this did not seem to result in a training set in which the parasite features of the validation set were well-represented. It is expected that when a larger volume of AiDx data is used in retraining, the classifier can reach the same level of performance as was achieved on the Rajaraman validation set.

Due to low precision of the network on the AiDx data, parasiteamia estimates for none of the images on which the network was evaluated were within 25 % of the correct value, the estimates were all overinflated, by 50.8 % on average, with one outlier where the estimate was 204 % higher. Here, retraining did improve performance, which can be attributed to the fact that a larger sample of non-infected cells than infected cells was available for re-training. For

the image that was used for validation, the parasitaemia estimate was 71 % too high before retraining the network, and 32 % after, which suggests that by retraining further, on added data, the predicted infection levels would eventually converge to the correct ones.

A binary classification strategy was used here, in which all objects were labelled as either being uninfected cells, or parasitised cells. A third class could be added for objects that were not cells. Given the high performance of the segmentation algorithm, this was not done here, however, doing so would increase the robustness of the predictions when poor results are produced in the segmentation step.

A transfer learning strategy was adopted in the design of the classifier, using the pre-trained VGG-16 model as a feature extractor. This option was chosen because of the high performance of this network on the ImageNet database and in other transfer learning image classification tasks found in literature. However, many other pre-trained models such as Xception, Resnet50 and Densenet201 are available, that have also shown good performance on these classification tasks [89]. Training a neural network based on one of these models, or selecting only the layers of these models that are the most effective for our specific task to build a more efficient ensemble model to use as feature extractor, might improve performance and would be an interesting topic for future research.

It is also possible to train a neural network from scratch on malaria image data, which would result in only extracting features that are useful for this specific task and remove redundancies in the network. However, previous work done on this subject has shown that in general, to extract robust image features of the same or better quality as the ones available in pre-trained networks, larger amounts of training data than are available currently would be needed. Networks trained from scratch on the currently available Rajaraman database, are not likely to generalise well to new input data [68, 69, 90]. The segmentation strategy proposed here, combined with the simple script that was developed to manually determine their ground truth infection status and divide these segmented objects over classes, could be applied towards efficiently generating a large malaria infected erythrocyte database, which would make training from scratch a more viable option in future research.

The ground truths for both the Rajaraman data and the AiDx data that were used, were found to be debatable. Rajaraman et al. reported that their ground truth had been determined by an expert slide reader at the Mahidol-Oxford Tropical Medicine Research Unit in Bangkok, Thailand [91], however when inspecting the images that had been labelled uninfected, some showed peripheral chromatin dots and even ring shapes, which suggests that these should have been labelled parasitised. It was unfortunately not doable within the constraints of this project to re-evaluate the ground truth for every single instance in this dataset. Given the good performance of the classifier trained on this data, on validation data from the same set, these inconsistencies were apparently small enough in quantity to not hinder performance significantly.

The ground truth for the AiDx dataset was verified by me, my colleague and dr. ir. T.E. Agbana (expert in this field). However, since the data was taken at lower magnification than is ordinarily the case, determining infection was not always straight-forward. Inspecting the results of the classifier, by looking at images such as the one shown in figure 4-11 raised some doubts about whether some of the objects that were labelled as FN in the validation were actually infected, or whether these were in fact TN. The opposite was also true, some objects labelled FP, might in fact be TP upon closer inspection. The best way to eliminate these ambiguities would be to have another independent medical expert manually annotate

the data.

Running the U-Net segmentation algorithm and the classification model consecutively, allows us to start with a blood smear image, and end up with an image such as the one shown in figure 4-8, in which predicted cell and parasite locations are indicated and a parasitemia estimate is printed. This takes 65 seconds on average on the laptop that was used (Intel Core i7 processor with 4 GB of RAM) for a single image, when operating in batch, the time taken per image is reduced. It is noted that the majority of running time was dedicated to operations such as saving all individual cell images to a directory, which can be eliminated when running consecutively. Predictions by the segmentation model took on average 6.045 seconds for a full image (20 image tiles), while predicting on the erythrocytes by the classification network took 59 ms per cell, so on average 21.24 seconds for a full image which contains 400 cells.

The WHO assessed that a human microscopist can realistically only read 30-40 slides a day, and that long hours of continuous reading result in fatigue, which can significantly reduce the accuracy of reading [5]. This is where this algorithm can make real impact, being able to read at least 1152 FOV a day on a consumer laptop without need for pause. To be applicable in on field settings, with a total size of 72 Mb, the software could be loaded onto a external computing device or a smart phone. Alternatively, if mobile data coverage is extended to the target setting, this can be leveraged to send images, perform interpretation externally and send back the result, allowing for even faster interpretations.

The scope of this research was limited to the interpretation of *P. Falciparum* Giemsa stained thin blood smears. This was chosen, as it is commonly the most widely accepted technique for the microscopic diagnosis of malaria, as well as the diagnostic method most limited by the time consumed to interpret the data. The choice to use only *P. Falciparum* infected samples was made based on availability, and because this is the most predominant species and most deadly species in the world. No attempts have thus been made in this project to attempt to automatically distinct between species, we highly recommend this as an area for future research.

Furthermore, as was discussed in chapter 1, other malaria microscopy methods are also available in practice, which could also benefit from automation, such as fluorescence microscopy, or have been proposed in literature, such as multispectral microscopy with unstained blood films. For the latter, some preliminary work was done during this project to design an appropriate classifier architecture for multi-spectral images. However, as no such data was available, this was not tested, and is therefore recommended as a topic for future work.

Chapter 6

Conclusion

The aim of this work was to contribute to the development of malaria diagnostic methods suitable for use in situ. Through review of the literature on diagnostic methods in chapter 1, a list of requirements for such a diagnostic method was presented in section 1-2. By combining requirements 2 and 3; the ability to determine parasitaemia counts and identify parasite species and stage, with requirement 5; the wish for minimal skill and labour needed to interpret the test, we arrived at researching the possibilities of automating the interpretation of Giemsa stained microscopy of thin blood films.

Through review of previous work on this subject, we arrived at the use of neural networks a promising technique for automated image interpretation. We chose to investigate specifically the interpretation of low magnification images, based on image data produced by a microscope that is currently in development at AiDx, which is portable and low cost and thus meets the 4th requirement we set for a novel diagnostic test.

This provided the motivation for our main research question;

To what extent can neural networks be applied towards eliminating the need for trained experts in the interpretation of low magnification Giemsa stained thin blood smears for malaria diagnostics?

To investigate this, two methods to segment erythrocytes from blood films were proposed, which allow us to estimate a cell count and produce objects which can later be classified.

The threshold based segmentation method proposed performed well on standard 100× magnification images (sensitivity 0.976), but performance was not as good on the AiDx images (sensitivity 0.895). Since only erythrocytes that are found are passed on the classifier, to obtain accurate diagnostics with this method, a larger volume of image data would be needed. The U-net based segmentation method showed excellent performance on the AiDx data; sensitivity, precision and F_1 -score were all 0.998. This method localises almost all present erythrocytes, and can therefore be used to estimate cell count vary accurately.

To classify the erythrocyte objects, a transfer learning strategy was proposed, which allowed us to exploit the robust general image features learned by the pre-trained VGG-16 network. A single fully connected layer was added to this architecture to efficiently predict correct

activations for these features. This classifier was trained and validated on a publicly available database of segmented erythrocyte objects, performance on this data was adequate; sensitivity was 0.984, specificity was 0.947, PPV was 0.949 and F_1 -score was 0.966.

The trained network did not perform as well on erythrocytes segmented from the AiDx images; sensitivity on this data was 0.795, specificity was 0.915, PPV 0.477 and F_1 -score was 0.636. Parasitemia estimates produced by this classifier were $1,5 \times$ the actual parasitemia of on average.

Improvements on performance measures were achieved by retraining the network on AiDx data; specificity increased to 0.948, PPV to 0.588 and the parasitemia estimate on the validation image was 39 % closer to the ground truth. However, since only one image was available to validate this retrained model, no definitive conclusions can be drawn.

The first requirement we set for a diagnostic method, was to have a specificity of $\geq 90\%$, which was achieved, and a detection limit of 50 parasites / μL of blood. To obtain that detection limit with the proposed method, 250 thin smear AiDx images would have to be analysed, which is theoretical possible since analysis is fully automated. However, the low precision did not allow us to accurately determine parasitemia, and no attempts were made to identify parasite species and stage.

We can therefore conclude that the proposed CNN-based methods can not fully eliminate the need for trained experts in the interpretation of low magnification Giemsa stained thin blood smears for malaria diagnostics. However, automatically determining an accurate cell count, as was done with the U-Net based segmentation method, is a big step in the right direction. Furthermore, when the results of the classification are presented to an expert in the visual way that was shown here, this expert can easily determine the true infection status of the objects predicted as infected, to correct for the low precision. This would greatly reduce the number of cells that need to be evaluated. We therefore believe that the method proposed can contribute to reducing the diagnostic burden, and increasing the availability of malaria diagnostics globally.

When the portable microscope is developed further, it can be used to obtain additional blood slide images, which can be applied towards improving the performance of the proposed neural networks. In future, a simple computing device that runs the proposed algorithms, such as a Raspberry Pi, can be integrated into the microscope design, to automatically execute the full diagnostic procedure in one integrated device.

Bibliography

- [1] World Health Organization, *World malaria report 2019*. 2019.
- [2] Center for Disease Control and Prevention, “Malaria - Image Gallery,” <https://www.cdc.gov/dpdx/malaria/>, 2019.
- [3] Center for Disease Control and Prevention, “Comparison of the Plasmodium Species Which Cause Human Malaria,” 2013.
- [4] M. Poostchi, K. Silamut, R. J. Maude, S. Jaeger, and G. Thoma, “Image analysis and machine learning for detecting malaria,” *Translational Research*, 2018.
- [5] World Health Organization, *Malaria microscopy quality assurance manual – Ver. 2*. 2016.
- [6] Center for Disease Control and Prevention, “Parasites - Malaria,” <https://www.cdc.gov/parasites/malaria/>, 2020.
- [7] J. G. Breman, M. S. Alilio, and A. Mills, “Conquering the intolerable burden of malaria: What’s new, what’s needed: A summary,” *American Journal of Tropical Medicine and Hygiene*, vol. 71, no. 2 SUPPL., pp. 1–15, 2004.
- [8] N. Tangpukdee, C. Duangdee, P. Wilairatana, and S. Krudsood, “Malaria diagnosis: A brief review,” *Korean Journal of Parasitology*, vol. 47, no. 2, pp. 93–102, 2009.
- [9] World Health Organization, *Guidelines for the Treatment of Malaria*. 3 ed., 2015.
- [10] K. O. Mfuh, O. A. Achonduh-Atijegbe, O. N. Bekindaka, L. F. Esemu, C. D. Mbakop, K. Gandhi, R. G. Leke, D. W. Taylor, and V. R. Nerurkar, “A comparison of thick-film microscopy, rapid diagnostic test, and polymerase chain reaction for accurate diagnosis of Plasmodium falciparum malaria,” *Malaria Journal*, vol. 18, mar 2019.
- [11] G. M. B. B. S. Dipti, C. A. Kumar, S. Baveja, and &. Head, “Comparative Staining Methods for Microscopic Diagnosis of Malaria,” *Paripex - Indian Journal Of Research*, vol. 5, no. 8, pp. 236–237, 2016.

- [12] D. C. Warhurst and J. E. Williams, "Laboratory diagnosis of malaria," *Journal of Clinical Pathology*, vol. 49, no. 7, pp. 533–538, 1996.
- [13] S. Sathpathi, A. K. Mohanty, P. Satpathi, S. K. Mishra, P. K. Behera, G. Patel, and A. M. Dondorp, "Comparing Leishman and Giemsa staining for the assessment of peripheral blood smear preparations in a malaria-endemic region in India," *Malaria Journal*, vol. 13, pp. 1–5, dec 2014.
- [14] S. Shillcutt, C. Morel, C. Goodman, P. Coleman, D. Bell, C. J. Whitty, and A. Mills, "Cost-effectiveness of malaria diagnostic methods in sub-Saharan Africa in an era of combination therapy," *Bulletin of the World Health Organization*, vol. 86, no. 2, pp. 101–110, 2008.
- [15] D. Payne, "Use and limitations of light microscopy for diagnosing malaria at the primary health care level," *Bulletin of the World Health Organization*, vol. 66, no. 5, pp. 621–626, 1988.
- [16] C. Ohrt and D. Tang, "Impact of microscopy error on protective efficacy estimates in malaria prevention trials," *Clinical Pharmacology and Therapeutics*, vol. 65, no. 2, p. 134, 1999.
- [17] A. Vasiman, J. R. Stothard, and I. I. Bogoch, "Mobile Phone Devices and Handheld Microscopes as Diagnostic Platforms for Malaria and Neglected Tropical Diseases (NTDs) in Low-Resource Settings: A Systematic Review, Historical Perspective and Future Outlook," *Advances in Parasitology*, vol. 103, pp. 151–173, jan 2019.
- [18] T. E. Agbana, J. C. Diehl, F. Van Pul, S. M. Khan, V. Patlan, M. Verhaegen, and G. Vdovin, "Imaging & identification of malaria parasites using cellphone microscope with a ball lens," *PLoS ONE*, vol. 13, no. 10, pp. 1–13, 2018.
- [19] C. Wongsrichanalai, J. Pornsilapatip, V. Namsiripongpun, H. K. Webster, A. Luccini, P. Pansamdang, H. Wilde, and M. Prasittisuk, "Acridine orange fluorescent microscopy and the detection of malaria in populations with low-density parasitemia," *American Journal of Tropical Medicine and Hygiene*, vol. 44, no. 1, pp. 17–20, 1991.
- [20] M. T. Makler, L. K. Ries, J. Ries, R. J. Horton, and D. J. Hinrichs, "Detection of *Plasmodium falciparum* infection with the fluorescent dye, benzothiocarboxypurine," *American Journal of Tropical Medicine and Hygiene*, vol. 44, pp. 11–16, jan 1991.
- [21] G. O. Adeoye and I. C. Nga, "Comparison of Quantitative Buffy Coat technique (QBC) with Giemsa-stained thick film (GTF) for diagnosis of malaria," *Parasitology International*, vol. 56, no. 4, pp. 308–312, 2007.
- [22] J. Kevin Baird, T. R. Purnomo, and T. R. Jones, "Diagnosis of malaria in the field by fluorescence microscopy of QBC® capillary tubes," *Transactions of the Royal Society of Tropical Medicine and Hygiene*, vol. 86, no. 1, pp. 3–5, 1992.
- [23] A. Benito, J. Roche, R. Molina, C. Amela, and J. Alvar, "Application and evaluation of QBC malaria diagnosis in a holoendemic area.," *Applied parasitology*, vol. 35, pp. 266–272, nov 1994.

-
- [24] A. Moody, "Rapid Diagnostic Tests for Malaria Parasites," *Clinical Microbiology Reviews*, vol. 15, no. 1, pp. 66–78, 2002.
- [25] C. J. Whitty, M. Armstrong, and R. H. Behrens, "Self-testing for falciparum malaria with antigen-capture cards by travelers with symptoms of malaria," *American Journal of Tropical Medicine and Hygiene*, vol. 63, pp. 295–297, nov 2000.
- [26] P. Mbabazi, H. Hopkins, E. Osilo, M. Kalungu, P. Byakika-Kibwika, and M. R. Kamya, "Accuracy of two malaria rapid diagnostic tests (RDTS) for initial diagnosis and treatment monitoring in a high transmission setting in Uganda," *American Journal of Tropical Medicine and Hygiene*, vol. 92, pp. 530–536, mar 2015.
- [27] S. P. Johnston, N. J. Pieniazek, M. V. Xayavong, S. B. Slemenda, P. P. Wilkins, and A. J. Da Silva, "PCR as a confirmatory technique for laboratory diagnosis of malaria," *Journal of Clinical Microbiology*, vol. 44, pp. 1087–1089, mar 2006.
- [28] B. Morassin, R. Fabre, A. Berry, and J. F. Magnaval, "One year's experience with the polymerase chain reaction as a routine method for the diagnosis of imported malaria," *American Journal of Tropical Medicine and Hygiene*, vol. 66, no. 5, pp. 503–508, 2002.
- [29] T. Häscheid and M. P. Grobusch, "How useful is PCR in the diagnosis of malaria?," *Trends in Parasitology*, vol. 18, pp. 395–398, sep 2002.
- [30] Y. Mori and T. Notomi, "Loop-mediated isothermal amplification (LAMP): A rapid, accurate, and cost-effective diagnostic method for infectious diseases," *Journal of Infection and Chemotherapy*, vol. 15, no. 2, pp. 62–69, 2009.
- [31] D. H. Paris, M. Imwong, A. M. Faiz, M. Hasan, E. B. Yunus, K. Silamut, S. J. Lee, N. P. Day, and A. M. Dondorp, "Loop-mediated isothermal PCR (LAMP) for the diagnosis of falciparum malaria," *American Journal of Tropical Medicine and Hygiene*, vol. 77, pp. 972–976, nov 2007.
- [32] J. Sirichaisinthop, S. Buates, R. Watanabe, E. T. Han, W. Suktawonjaroenpon, S. Krasaesub, S. Takeo, T. Tsuboi, and J. Sattabongkot, "Short report: Evaluation of loop-mediated isothermal amplification (LAMP) for malaria diagnosis in a field setting," *American Journal of Tropical Medicine and Hygiene*, vol. 85, pp. 594–596, oct 2011.
- [33] D. L. Omucheni, K. A. Kaduki, W. D. Bulimo, and H. K. Angeyo, "Application of principal component analysis to multispectral-multimodal optical image analysis for malaria diagnostics," *Malaria Journal*, vol. 13, pp. 1–11, dec 2014.
- [34] J. Klossa, B. Wattelier, T. Happillon, D. Toubas, L. de Laulanie, V. Untereiner, P. Bon, and M. Manfait, "Quantitative phase imaging and Raman micro-spectroscopy applied to Malaria," *Diagnostic Pathology*, vol. 8, p. S42, sep 2013.
- [35] V. Wongchotigul, N. Suwanna, S. Krudsood, D. Chindanond, S. Kano, N. Hanaoka, Y. Akai, Y. Maekawa, S. Nakayama, S. Kojima, and S. Looareesuwan, "The use of flow cytometry as a diagnostic test for malaria parasites," *Southeast Asian Journal of Tropical Medicine and Public Health*, vol. 35, no. 3, pp. 552–559, 2004.

- [36] P. F. Scholl, D. Kongkasuriyachai, P. A. Demirev, A. B. Feldman, J. S. Lin, D. J. Sullivan, and N. Kumar, "Rapid detection of malaria infection in vivo by laser desorption mass spectrometry," *American Journal of Tropical Medicine and Hygiene*, vol. 71, pp. 546–551, nov 2004.
- [37] F. Chen, B. R. Flaherty, C. E. Cohen, D. S. Peterson, and Y. Zhao, "Direct detection of malaria infected red blood cells by surface enhanced Raman spectroscopy," *Nanomedicine: Nanotechnology, Biology, and Medicine*, vol. 12, pp. 1445–1451, aug 2016.
- [38] H. T. Ngo, N. Gandra, A. M. Fales, S. M. Taylor, and T. Vo-Dinh, "Sensitive DNA detection and SNP discrimination using ultrabright SERS nanorattles and magnetic beads for malaria diagnostics," *Biosensors and Bioelectronics*, vol. 81, pp. 8–14, jul 2016.
- [39] C. Doderer, A. Heschung, P. Guntz, J. P. Cazenave, Y. Hansmann, A. Senegas, A. W. Pfaff, T. Abdelrahman, and E. Candolfi, "A new ELISA kit which uses a combination of Plasmodium falciparum extract and recombinant Plasmodium vivax antigens as an alternative to IFAT for detection of malaria antibodies," *Malaria Journal*, vol. 6, pp. 1–8, feb 2007.
- [40] J. W. Park, S. B. Yoo, J. H. Oh, J. S. Yeom, Y. H. Lee, Y. Y. Bahk, Y. S. Kim, and K. J. Lim, "Diagnosis of vivax malaria using an IgM capture ELISA is a sensitive method, even for low levels of parasitemia," *Parasitology Research*, vol. 103, pp. 625–631, aug 2008.
- [41] A. S. Abdul Nasir, M. Y. Mashor, and Z. Mohamed, "Segmentation based approach for detection of malaria parasites using moving k-means clustering," in *2012 IEEE-EMBS Conference on Biomedical Engineering and Sciences, IECBES 2012*, pp. 653–658, 2012.
- [42] N. Linder, R. Turkki, M. Walliander, A. Mårtensson, V. Diwan, E. Rahtu, M. Pietikäinen, M. Lundin, and J. Lundin, "A malaria diagnostic tool based on computer vision screening and visualization of Plasmodium falciparum candidate areas in digitized blood smears," *PLoS ONE*, vol. 9, p. e104855, aug 2014.
- [43] S. S. Savkare and S. P. Narote, "Automated system for malaria parasite identification," in *Proceedings - 2015 International Conference on Communication, Information and Computing Technology, ICCICT 2015*, Institute of Electrical and Electronics Engineers Inc., feb 2015.
- [44] L. Rosado, J. M. Da Costa, D. Elias, and J. S. Cardoso, "Automated Detection of Malaria Parasites on Thick Blood Smears via Mobile Devices," in *Procedia Computer Science*, vol. 90, pp. 138–144, Elsevier B.V., jan 2016.
- [45] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Pearson, 3 ed., 2008.
- [46] E. Y. Lam, "Combining gray world and retinex theory for automatic white balance in digital photography," in *Proceedings of the International Symposium on Consumer Electronics, ISCE*, pp. 134–139, 2005.
- [47] D. K. Das, A. K. Maiti, and C. Chakraborty, "Automated system for characterization and classification of malaria-infected stages using light microscopic images of thin blood smears," *Journal of Microscopy*, vol. 257, no. 3, pp. 238–252, 2015.

-
- [48] R. Tomari, W. N. W. Zakaria, M. M. A. Jamil, F. M. Nor, and N. F. N. Fuad, "Computer aided system for red blood cell classification in blood smear image," in *Procedia Computer Science*, vol. 42, pp. 206–213, Elsevier B.V., jan 2014.
- [49] D. Anggraini, A. S. Nugroho, C. Pratama, I. E. Rozi, V. Pragesjvara, and M. Gunawan, "Automated status identification of microscopic images obtained from malaria thin blood smears using Bayes decision: A study case in Plasmodium falciparum," in *ICACISIS 2011 - 2011 International Conference on Advanced Computer Science and Information Systems, Proceedings*, pp. 347–352, 2011.
- [50] S. S. Savkare, A. S. Narote, and S. P. Narote, "Automatic blood cell segmentation using K-Mean clustering from microscopic thin blood images," in *ACM International Conference Proceeding Series*, vol. 21-24-Sept, (New York, New York, USA), pp. 8–11, Association for Computing Machinery, sep 2016.
- [51] M. C. Mushabe, R. Dendere, and T. S. Douglas, "Automated detection of malaria in Giemsa-stained thin blood smears," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 3698–3701, 2013.
- [52] J. M. Sharif, M. F. Miswan, M. A. Ngadi, M. S. H. Salam, and M. M. B. A. Jamil, "Red blood cell segmentation using masking and watershed algorithm: A preliminary study," in *2012 International Conference on Biomedical Engineering, ICoBE 2012*, pp. 258–262, 2012.
- [53] L. H. Zou, J. Chen, J. Zhang, and N. García, "Malaria cell counting diagnosis within large field of view," in *Proceedings - 2010 Digital Image Computing: Techniques and Applications, DICTA 2010*, pp. 172–177, 2010.
- [54] S. Shuleenda Devi, S. Alam Sheikh, and R. Hussain Laskar, "Erythrocyte Features for Malaria Parasite Detection in Microscopic Images of Thin Blood Smear: A Review," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 4, no. 2, p. 34, 2016.
- [55] F. B. Tek, A. G. Dempster, and I. Kale, "Parasite detection and identification for automated thin blood film malaria diagnosis," *Computer Vision and Image Understanding*, vol. 114, no. 1, pp. 21–32, 2010.
- [56] S. S. Savkare and S. P. Narote, "Automatic Detection of Malaria Parasites for Estimating Parasitemia," *Narote International Journal of Computer Science and Security (IJCSS)*, no. 5, p. 310, 2011.
- [57] D. K. Das, M. Ghosh, M. Pal, A. K. Maiti, and C. Chakraborty, "Machine learning approach for automated screening of malaria parasite using light microscopic images," *Micron*, vol. 45, pp. 97–106, feb 2013.
- [58] D. K. Das, R. Mukherjee, and C. Chakraborty, "Computational microscopic imaging for malaria parasite detection: A systematic review," *Journal of Microscopy*, vol. 260, no. 1, pp. 1–19, 2015.

- [59] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, pp. 84–90, jun 2017.
- [60] T. Chen and H. Chen, “Universal Approximation to Nonlinear Operators by Neural Networks with Arbitrary Activation Functions and Its Application to Dynamical Systems,” *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 911–917, 1995.
- [61] X. Glorot, A. Bordes, and Y. Bengio, “Deep sparse rectifier neural networks,” *Journal of Machine Learning Research*, vol. 15, pp. 315–323, 2011.
- [62] F. Agostinelli, M. Hoffman, P. Sadowski, and P. Baldi, “Learning activation functions to improve deep neural networks,” in *3rd International Conference on Learning Representations, ICLR 2015 - Workshop Track Proceedings*, pp. 1–9, 2015.
- [63] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, “Deconvolutional networks,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2528–2535, IEEE, 2010.
- [64] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” mar 2016.
- [65] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the thirteenth international conference on artificial intelligence and statistics.*, pp. 249–256, JMLR Workshop and Conference Proceedings, mar 2010.
- [66] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 1026–1034, 2015.
- [67] Y. Dong, Z. Jiang, H. Shen, W. David Pan, L. A. Williams, V. V. Reddy, W. H. Benjamin, and A. W. Bryan, “Evaluations of deep convolutional neural networks for automatic identification of malaria infected cells,” *2017 IEEE EMBS International Conference on Biomedical and Health Informatics, BHI 2017*, pp. 101–104, 2017.
- [68] S. Rajaraman, S. K. Antani, M. Poostchi, K. Silamut, M. A. Hossain, R. J. Maude, S. Jaeger, and G. R. Thoma, “Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images,” *PeerJ*, vol. 2018, no. 4, 2018.
- [69] G. P. Gopakumar, M. Swetha, G. Sai Siva, and G. R. Sai Subrahmanyam, “Convolutional neural network-based malaria diagnosis from focus stack of blood smear images acquired using custom-built slide scanner,” *Journal of Biophotonics*, vol. 11, no. 3, pp. 1–18, 2018.
- [70] T. Wollmann, M. Gunkel, I. Chung, H. Erfle, K. Rippe, and K. Rohr, “GRUU-Net: Integrated convolutional and gated recurrent neural network for cell segmentation,” *Medical Image Analysis*, vol. 56, pp. 68–79, 2019.
- [71] M. Delgado-Ortet, A. Molina, S. Alférez, J. Rodellar, and A. Merino, “A Deep Learning Approach for Segmentation of Red Blood Cell Images and Malaria Detection,” *Entropy*, vol. 22, no. 6, p. 657, 2020.

-
- [72] R. Girshick, “Fast R-CNN object detection with Caffe,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015.
- [73] C. Mehanian, M. Jaiswal, C. Delahunt, C. Thompson, M. Horning, L. Hu, S. McGuire, T. Ostbye, M. Mehanian, B. Wilson, C. Champlin, E. Long, S. Proux, D. Gamboa, P. Chiodini, J. Carter, M. Dhorda, D. Isaboke, B. Ogutu, W. Oyibo, E. Villasis, K. M. Tun, C. Bachman, and D. Bell, “Computer-Automated Malaria Diagnosis and Quantitation Using Convolutional Neural Networks,” in *Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017*, vol. 2018-Janua, pp. 116–125, 2017.
- [74] A. Loddo, C. Di Ruberto, M. Kocher, and G. Prod’hom, “Mp-idb: The malaria parasite image database for image processing and analysis,” in *Processing and Analysis of Biomedical Information*, vol. 11379, pp. 57–65, Springer, 2019.
- [75] M. D. Abràmoff, P. J. Magalhães, and S. J. Ram, “Image processing with imageJ,” *Biophotonics International*, vol. 11, no. 7, pp. 36–41, 2004.
- [76] A. M. Reza, “Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement,” *Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology*, vol. 38, no. 1, pp. 35–44, 2004.
- [77] Otsu and N., “A threshold selection method from gray-level histograms,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [78] E. Dougherty, S. Beucher, and F. Meyer, “The Morphological Approach to Segmentation: The Watershed Transformation,” in *Mathematical Morphology in Image Processing*, pp. 433–481, CRC Press, jan 1993.
- [79] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, eds.), (Cham), pp. 234–241, Springer International Publishing, 2015.
- [80] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, “UNet++: A nested U-Net architecture for medical image segmentation,” in *arXiv* (D. Stoyanov, Z. Taylor, G. Carneiro, T. Syeda-Mahmood, A. Martel, L. Maier-Hein, J. M. R. S. Tavares, A. Bradley, J. P. Papa, V. Belagiannis, J. C. Nascimento, Z. Lu, S. Conjeti, M. Moradi, H. Greenspan, and A. Madabhushi, eds.), (Cham), pp. 3–11, Springer International Publishing, 2018.
- [81] F. Chollet and Others, “Keras.” <https://keras.io>, 2015.
- [82] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,” 2016.

- [83] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, pp. 1–18, 2012.
- [84] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis," in *Icdar*, 2003.
- [85] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–14, 2015.
- [86] J. Deng, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, p. 248, 2009.
- [87] M. Lin, Q. Chen, and S. Yan, "Network in network," in *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, pp. 1–10, 2014.
- [88] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 2019.
- [89] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [90] Z. Liang, A. Powell, I. Ersoy, M. Poostchi, K. Silamut, K. Palaniappan, P. Guo, M. A. Hossain, A. Sameer, R. J. Maude, J. X. Huang, S. Jaeger, and G. Thoma, "CNN-based image analysis for malaria diagnosis," in *Proceedings - 2016 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2016*, pp. 493–496, IEEE, 2017.
- [91] S. Rajaraman, S. Jaeger, and S. K. Antani, "Performance evaluation of deep neural ensembles toward malaria parasite detection in thin-blood smear images," *PeerJ*, vol. 7, p. e6977, may 2019.

Glossary

List of Acronyms

WHO	World Health Organization
RDTs	Rapid Diagnostic Tests
FOV	Field of View
QBC	Quantitative Buffy Coat
AO	acridine orange
PCR	Polymerase Chain Reaction
LAMP	loop-mediated isothermal amplification
SERS	surface-enhanced Raman scattering
SVM	Support Vector Machine
ANN	Artificial Neural Network
INLSVRC	ImageNet Large Scale Visual Recognition Challenge
CNN	Convolutional Neural Network
ReLU	Rectified Linear Unit
MSE	Mean Squared Error
CNN	Convolutional Neural Network
CLAHE	Contrast Limited Adaptive Histogram Equalization
TP	true positive
FP	false positive
FN	false negative

TN	true negative
TPR	true positive rate
TNR	true negative rate
PPV	positive predictive value
ROC	Receiver Operating Characteristic
IoU	Intersection over Union