MobiMal: Machine Learning For Malaria Diagnosis - Draft

Daniel Collins
University of Cape Town
Cape Town, South Africa
danielcollins96@gmail.com

ABSTRACT

Malaria is a major health problem that affects many parts of the world, where infection can quickly lead to death if victims go untreated. In most areas the most reliable form of malaria diagnosis is in the microscopic examination of a patients stained blood film performed by a trained microscopist [15]. This work presents a tool which uses two different machine learning methods, a Support Vector Machine and an Artificial Neural Network, to assist in the analysis of blood stains for malaria diagnosis. After using a series of preprocessing methods to assist in the identification of individual blood cell images, both algorithms achieve accuracy scores for the classification of malaria infected blood cells of above 96%.

KEYWORDS

Image Segmentation, Machine Learning, Erythrocytes, Malaria, Support Vector Machine, Artificial Neural Network, Pre-processing

1 INTRODUCTION

Automatic malaria diagnosis tools using computer vision have been studied and developed extensively in the past years [7][14]. Initially, studies focused on the detection of parasites within blood samples using image segmentation methods. More recently, with the rise of machine learning methods and techniques, accurate tools which can perform classification of the malaria parasite's life-stage and species have been developed [14]. These automatic diagnosis tools allow for the possibility of improved treatment quality and speed [7].

The importance of effectively implementing these tools is justified by the global health problem that malaria presents. In 2015 the number of global malaria cases was an estimated 214 million, with 88% of these occurring in the WHO African Region. During that year there were an estimated 438 000 malaria deaths [17]. Furthermore, the gold standard for malaria diagnosis remains examination of peripheral thick or thin blood smears (light microscopy) due to its low cost and resource requirements [25]. However, diagnosis which relies heavily upon blood smear examination can lead to treatment delays and in turn cause higher case fatality ratios [11]. Developing tools which can reduce the error associated with and the time required for these examinations is therefore vital to better combating the disease.

The challenges of automatic diagnosis tools are largely due to the wide scope of the diagnosis task. To most correctly and effectively diagnose a person, a blood sample must be examined to determine the species of malaria present, the life-stages present and the number of erythrocytes (red blood cells) which are infected [25]. In the context of computer vision, identifying the objects of interest becomes harder the more varied the structures of those objects are. In *Figure 1* the wide range of structures that must be accounted for can be seen. Further potential problems arise with the overlapping of blood cells, which could cause infected blood cells to be overlooked, weakening the integrity of the diagnosis. Examples of overlapping cells can be found in *Figure 1*. Another challenge facing automatic diagnosis tools is the removal of artefacts that don't contribute to the diagnosis of a blood sample. These artefacts could be caused by low quality blood film preparation, and include fungi, dirt and cell debris [6] .

The aim of this paper is to find a suitable machine learning algorithm and set of input features that will work as a successful automatic malaria diagnosis/diagnosis assistance tool. While the pre-processing methods used play an integral part in the overall success of the solution, the main focus will be on the development on a set of input features for the algorithms, and the configuration of the algorithms used. This work aims to implement a solution which builds upon previously published works. To achieve this we developed a tool that uses two machine learning algorithms to classify red blood cells according to whether or not they are infected with *Plasmodium falciparum*. The choice of species is due to *Plasmodium falciparum* being responsible for over 90% of malaria infections in South Africa [11]. The following two machine learning algorithms are used:

(1) A Multi-layer perceptron neural network model which is a supervised machine learning algorithm. This model takes as input a set of features and returns the output value corresponding to these features, which in the context of this project will be true or false diagnosis per identified red blood cell.
(2) A Support Vector Machine which is also a supervised learning algorithm. This method takes as input a set of features and transforms them into linearly separable data using a radial basis function (RBF). This separation of features allows the model to classify unseen input features.

For such tools to be considered as viable replacements for current manual methods they must perform as well as or better than human analysis. The WHO sets the following standards for tier 1 (expert) examiners [16]:

(1) Detect parasites and classify their species within 90-100% accuracy
(2) Count the number of parasite to within 25% of the true count
(3) Examine 50 blood smears with the above accuracy in 4 hours (approximately 5 minutes per sample)

The proposed solution uses standards (2) and (3) as the base requirements for the tool to be considered viable.

The remainder of this paper is presented as follows: Section 2 presents an investigation into methods used in previously published papers, and provides an understanding of each of these methods.

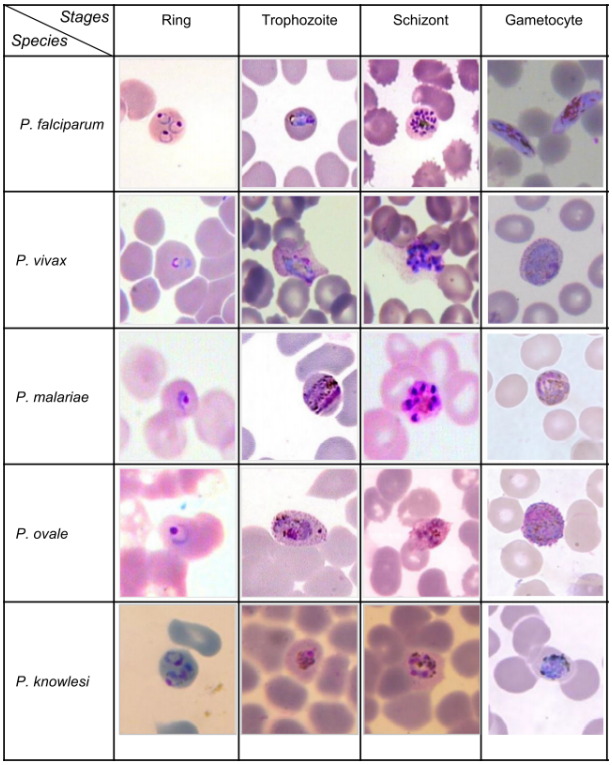


Figure 1: Malaria Plasmodium species and life stages in thin blood smears

Section 3 details the data used and how it was acquired, the design of the experiments performed, and how the results will be measured. Section 4 details the results of the experiments carried out. This is followed by Section 5 which interprets the results and then finally Section 6, which concludes the report by assessing the success of the research and suggests avenues for future research.

2 BACKGROUND AND RELATED WORK

As previously stated, automatic malaria diagnosis has been studied extensively in previous years. This section will explain concepts utilised in this project and examine approaches used in previous works. There are 2 steps in this solution, pre-processing and then classification.

2.1 Pre-Processing

The goal of preprocessing is to improve the image data and suppress distortions like noise, or enhance certain features that are needed for further processing [13]. This solution makes use of a variety of preprocessing methods applied sequentially to the blood smear images in order to make the individual blood cells more easily recognisable.

2.1.1 Greyscale Conversion. A grey-scale image uses a single colour channel for each pixel in an image, rather than the 3 colour channels (red, green and blue), required for colour images. Each pixel intensity is represented as a single number from 0 through to some maximum, typically 255 (1 byte).This process was used in [19] to reduce the size of the image, and in [12] to assist the image segmentation process used.

2.1.2 Contrast enhancement. The contrast of the image is enhanced through the use adaptive histogram equalisation. The adaptive nature of this method allows for it to effectively enhance contrast in images without prior knowledge of any of the image properties. This is appropriate for blood smear microscopy images which can have widely varying properties.

2.1.3 Morphological Filtering. Some images may contain imperfections like noise. Morphological processing attempts to remove these imperfections by taking the structure and form of the image into account [26]. This process probes an image with a small shape of pixels called a structuring element, which then according to the goal of the filter, will replace pixels that match the requirements of the structuring element. This then creates a new eroded or dilated image [26]. This method was used to smooth areas of the image in [23] [22].

2.1.4 Binary Thresholding. Binary thresholding is the process of reducing a greyscale image into a binary image. This project uses Otsu's Method [18], which finds the grey pixel value that can best separate the pixel values of an image into two classes, which in the context of this work is the foreground (red blood cells) and the background (the rest of the image).

2.2 Classification

Machine learning is a process that includes the organisation of new knowledge into general, effective representations that can be used to recognise similar instances of data. [21]. If an algorithm is given instances with known labels that correspond to the correct outputs, then this learning is categorised as *supervised* [10]. If the data provided is unlabelled then this is called *unsupervised.* Both classification approaches used in this work are supervised machine learning methods.

2.2.1 Support Vector Machine. Support Vector Machines (SVMs) work to create the biggest separation of selected features in order to classify input instances. SVMs are are linear classifiers, but can can work with non-linear input if the data has been transformed into higher dimensions [24].This higher-dimensional space is also called a feature space [10]. The basic goal of an SVM is to find the hyper-plane that best separates classes in this new feature space [4]

Linder et al [8] divided candidate regions containing positive diagnosis features into 64x64 pixel image windows from which features were extracted and fed into an SVM. This reduced the amount of data the SVM would have to process per image. The sensitivity of this scheme for erythrocyte classification was 84.9%. Kumar Das et al [3] used an RBF kernel to transform the image data into linearly separable data and then fed that into an SVM for classification. This approach achieved a maximum accuracy of 83.5%.
A support vector machine was chosen as a method for the classification of infected blood cells due to its ability to separate features in a higher-dimensional space. This will enable us to extract metadata

from the samples analysed which will be used as the features to classify the blood cells.

2.2.2 Artificial Neural Network. When data instances are not linearly separable, an artificial neural network is needed to separate instances into the correct categories, or classes [10]. An ANN usually consists of three units: input units, where we input the processed data; output units, where the results of the processes are found; and and units in-between, known as hidden units or layers. A visual representation of this can be seen in Figure 2.

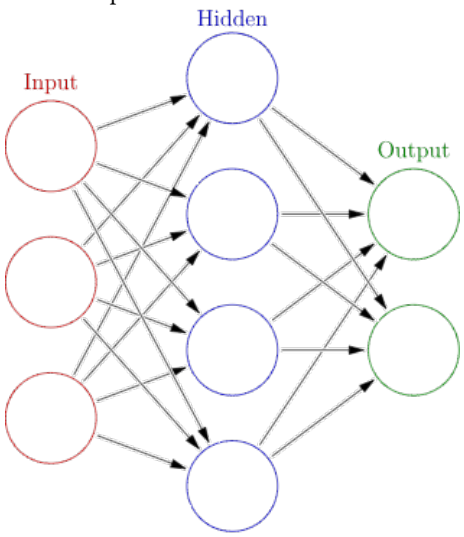


Figure 2: Typical visualisation of an artificial neural networks layers[1]

A three layer ANN classifier was used by Ross et al [20] to classify malaria infected blood cells with a sensitivity of 85%. Wijayarathna et al [5] found that a three layer ANN outperformed a four layer ANN by a 7.14% margin in the rate of accuracy for recognising malaria in blood samples. Amiga et al [9] used the RGB values of an image to train an ANN to classify the life stages of malaria present in a blood cell.

An ANN was chosen as a method for this work due to its ability to recognise patterns in large sets of features and generalise them to respond to unseen inputs. This is appropriate in this case because red blood cells often vary in texture. A rule of thumb according to Blum [2] is to set the size of the hidden layers between the size of the input layer and the output layer.

3 METHODOLOGY

In order to test which machine learning algorithm best classified red blood cells correctly, a set of experiments were designed to find the best results with the available data.

3.1 Overview of Experiment

Two sets of experiments were done to find the configurations and feature sets for the algorithms that achieved the highest accuracy. The first experiment aimed to find the set of features used as input and the variance window size that achieved the highest accuracy for the ANN. It also aimed to find the set of features and the window size that achieved the highest accuracy for the SVM. The features and variance windows are explained in section 3.6. The second part of the experiment aimed to compare the execution times and accuracies of the configurations of the separate algorithms from the first set of experiments.

3.2 Software and Hardware Used

The operating system used was Ubuntu LTS version 18, as downloading and installing software packages on this system is simple and free. The main libraries used were all Python 3 libraries and include Scikit-learn, Skicit-image and NumPy. Scikit-learn was used for the classification process, Scikit-image was used for preprocessing, and NumPy was used for array manipulation and other data processing. The hardware used for the experiments was a single Asus X550VX laptop with an Intel Core i7-6700HQ CPU @ 2.6GHz and 16GB RAM. The Nvidia Geforce GTX 950M on the device was not used during

3.3 Data Acquisition

The digital images used in this work were supplied by PathCare. Anonymised thin blood smears were prepared for malaria microscopy examination according to WHO standards. An expert malaria microscopist then took pictures of the smears with a light microscope at x100 magnification. All blood samples used were positively infected with *Plasmodium falciparum*, and the number of infected erythrocytes present in each sample was provided.

This research was approved by the University of Cape Town Faculty of Science Research Ethics Committee and PathCare for ethics clearance.

3.4 Dataset Augmentation

Due to not having the amount of data usually required to sufficiently train a machine learning model, a new dataset was created performing simple transformations on the original samples as was done by Linder et al.[8]. This dataset will be obtained by flipping each image, doubling the dataset, and then rotating these 2 images by 90°, 180° and 270°, which yields 8 times the original number of samples.

3.5 Pre-processing

The pre-processing procedure aims to make the separation of red blood cells from the rest of the image as accurate as possible. All parameters (eg. window size) specified were selected through an iterative process where different values were tested and the values which resulted in the most red blood cells being identified were chosen. The same concept was applied to the order in which these processes were applied to the images. The changes in the image during each pre-processing step are shown in Fig 3.

The images were first converted to grey-scale to make the image contain only 1 colour channel as in 3b. The contrast of the image was then enhanced using a 5x5 window to increase the difference in pixel values between the foreground and background of the image, shown in 3c. The image was then inverted to aid the next step. To invert an image each pixel value must be replaced by its maximum

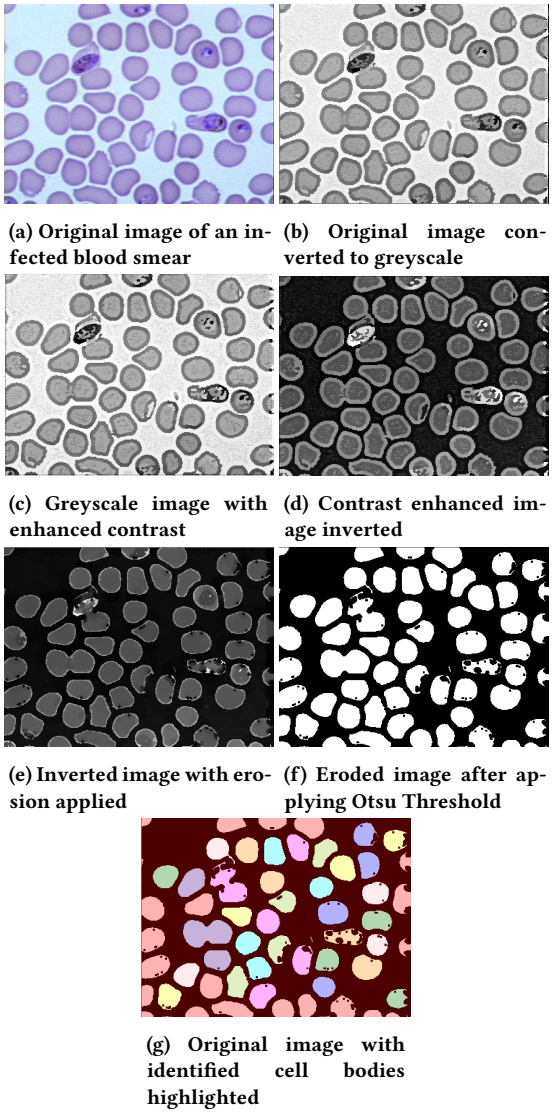
possible value minus its current value, as in 3d. A 2x2 erosion disk was then used to increase the space between red blood cells, and to smooth the image. Erosion achieves this as it enlarges dark regions and shrinks light regions, as shown in 3e. The Otsu threshold for the image is then calculated, and pixel values below this threshold are set to their minimum possible value, and pixel values above the threshold are set to their maximum possible value. This is shown in 3f. This is the last step in the segmentation of the foreground from the background of the image. The regions of connected pixels are then labelled and their dimensions recorded. Neighbouring pixels are considered connected when they have the same value. To demonstrate this, connected regions have been highlighted with colour in 3g.

The areas of the labelled regions are then calculated along with the mean μ and standard deviation σ. An interval of $(\mu - \sigma, \mu + \sigma)$ is then set, and regions whose area does not fall within this interval are excluded. This is to prevent bodies that are not red blood cells from being processed, since they are the wrong size. The remaining regions are then extracted and resized to 25x25 pixels.
Before the samples are used for training or testing, they have histogram equalisation applied to them to correct the varying exposure the samples may have from different lab conditions or equipment quality. These are the samples used in the classification process.

3.6 Features Used for Training and Testing

Two sets of data were used to test and train each algorithm individually in order to find which set would yield the highest accuracy. The first set of data was a 2D greyscale (2.1.1) representation of each image. The second set was the 3D images where the dimensions consisted of the red blue green (RGB) pixel values for each coordinate of the image. Each set of data then contains a varying amount of colour channels in an attempt to find the combination which yields the best results.
Instead of using individual pixel values as input features for the models as done in previous works, this work attempts to create an abstraction of the pixel values in an image. This feature is obtained by replacing the pixel values in the image with some multiple of the variance of the pixel values surrounding that pixel. The size of the window in which the variance is calculated will change during testing. This both reduces the number of input features and increases the ability of the algorithms to make generalisations of data. To make the feature extraction above applicable to our datasets containing more than 2 dimensions, the variance calculations will be done on each colour channel contained in an image and then reshaped into a 1D array for input into each algorithm. So for each "layer" of colour for the RGB representation of a sample, each value will be replaced with a multiple of the variance of the surrounding values in that layer.

This choice of input feature was motivated by the observation of the change in colour of infected blood cells during the blood smear preparation process. Once the staining process is complete, the *Plasmodium falciparum* present in the red blood cells is of a darker colour than the infected blood cell, causing it to stand out.



(a) Original image of an infected blood smear

(b) Original image converted to greyscale

(c) Greyscale image with enhanced contrast

(d) Contrast enhanced image inverted

(e) Inverted image with erosion applied

(f) Eroded image after applying Otsu Threshold

(g) Original image with identified cell bodies highlighted

Figure 3: Blood slide image during various pre-processing stages

This causes an increase in the variance of the pixel values surrounding the parasite. It is hypothesised that healthy blood cells will have a smaller variance of pixel values towards the centre of the blood cell.

The number of features in each sample is relative to the size of the window in which the variance is calculated according to the following formula, where *width* and *height* refer to the measurement in pixels of the sample image, w is the size in pixels of the sides of the window, and *channels* is the number of colour channels used:

$$features = (height - (w - 1)) * (width - (w - 1)) * channels \quad (1)$$

3.7 Testing

The samples used for training and classification are randomised with each iteration, but each iteration will have 300 healthy and 100 infected blood cells infected used for training, and then 100 healthy and 76 infected blood cells for classification. This is to prevent too few infected samples from being randomly chosen, which would decrease the networks ability to identify them.

3.7.1 Support Vector Machine. A standard Support Vector Machine using a radial basis function kernel was used. This set-up was not altered through the training and testing process.

3.7.2 Artificial Neural Network. The ANN used was comprised of 4 layers. One input layer accepting features, two hidden layers, and one output layer containing two neurons. The size of the first hidden layer will be set to half the number of input features for each network. The size of the second hidden layer will be set to half the size of the first hidden layer. This configuration was advised by Blum [2].

3.8 Evaluation

The accuracy of the methods will be calculated in terms of true positive (TP), false positive (FN), true negative (TN) and false negative (FN), as :

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

To compare the execution times and accuracies of the two algorithms appropriately, we record the average classification times and accuracies of the two configurations that achieved the highest accuracy score. The average execution time and accuracy is recorded over 20 iterations, along with the standard deviation. Both algorithms are trained with the dataset containing 400 samples, and then 176 samples are used for classification. The estimated time taken to classify 5000 red blood cells is also calculated and compared, as this amount is the guideline set by the WHO [16] for a diagnosis.

3.9 Statistical Analysis

To test for a significant difference in the mean execution times and accuracies of the two algorithms we will use the t test, which is used to test hypothesis for difference in means for small samples.

4 PERFORMANCE AND RESULTS

This section provides the tabulated results of the experiments carried out for both algorithms in various configurations. Each algorithms input features are graphically represented with a plot chart, followed by a statistical comparison of the accuracies and execution times.

4.1 ANN Results

The following tables contain the results of the ANN classification algorithm from the experiments carried out, showing the accuracy, along with the various window sizes, sizes of hidden layers, and the number of input features. This is followed by a plot chart showing the relationship between variance window size and classification accuracy for the ANN.

Window Size	No. Features	Hidden Layer Nodes	Accuracy	Standard Deviation
2x2	576	(864, 432)	93.438%	1.914%
3x3	529	(793, 396)	96.591%	1.333%
4x4	484	(726, 363)	95.881%	1.655%
5x5	441	(661, 330)	95.341%	2.305%
6x6	400	(600, 300)	94.091%	1.871%
7x7	361	(541, 270)	94.062%	1.508%
8x8	324	(486, 243)	93.778%	1.661%
9x9	289	(433, 216)	93.75%	1.867%
10x10	256	(384, 192)	93.324%	2.39%
11x11	225	(337, 168)	93.523%	2.479%
12x12	196	(294, 147)	93.097%	2.158%
13x13	169	(253, 126)	92.443%	1.894%
14x14	144	(216, 108)	92.528%	1.707%
15x15	121	(181, 90)	94.716%	1.557%
16x16	100	(150, 75)	94.148%	2.316%
17x17	81	(121, 60)	86.875%	2.609%

Table 1: ANN configurations and results using features extracted from equalised greyscale image values

Window Size	No. Features	Hidden Layer Nodes	Accuracy	Standard Deviation
2x2	1728	(864, 432)	91.306%	2.294%
3x3	1587	(793, 396)	94.090%	1.791%
4x4	1452	(726, 363)	95.596%	2.031%
5x5	1323	(661, 330)	95.965%	1.655%
6x6	1200	(600, 300)	95.000%	1.710%
7x7	1083	(541, 270)	95.653%	1.912%
8x8	972	(486, 243)	96.306%	1.939%
9x9	867	(433, 216)	95.681%	2.239%
10x10	768	(384, 192)	95.170%	2.020%
11x11	675	(337, 168)	94.630%	1.680%
12x12	588	(294, 147)	93.806%	2.488%
13x13	507	(253, 126)	93.806%	1.786%
14x14	432	(216, 108)	93.863%	1.613%
15x15	363	(181, 90)	95.482%	1.818%
16x16	300	(150, 75)	95.767%	1.157%
17x17	243	(121, 60)	94.318%	1.704%

Table 2: ANN configurations and results using features extracted from equalised RGB image values

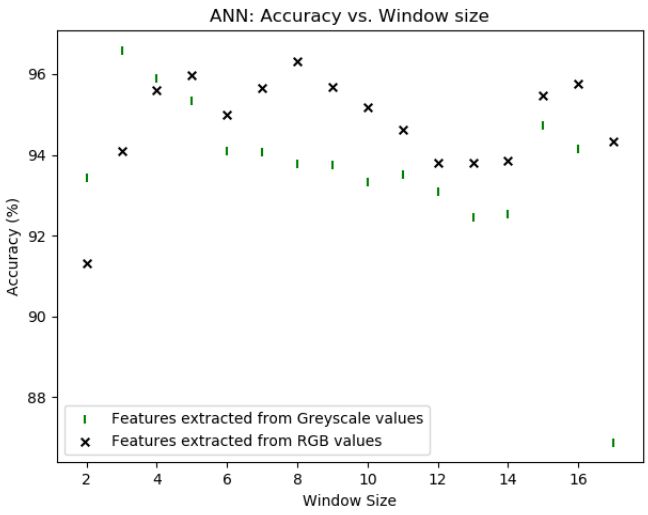


Figure 4: Plot chart of the ANN accuracies using features extracted from varying window sizes an RGB image and a greyscale image

There is a correlation coefficient of r=0.5865 between the number of features and the classification accuracy in Table 1. This suggests a strong relationship between the classification accuracy and the number of features used as input. The correlation coefficient between the variance window size and the accuracy is -0.6167, which suggests a strong negative relationship between window size and accuracy. This can be seen in figure 4, with accuracy slightly decreasing with an increase in window size.

The correlation coefficients between the accuracy and window size for the ANN using RGB extracted values in Table 2 is very weak (<0.2), which is also seen in figure 4, where the accuracy values seem to be oscillating up and down with an increase in window size.

There is a slight difference in the average performances of the two features sets used to train and test the ANN, with the ANN using RGB values averaging at 94.777% accuracy ahead of the ANN using greyscale values at 93.0% (t-test, t=1.61257, p=0.117). But this is not a large enough difference to be considered significant.

Window Size	No. Features	Accuracy	Standard Deviation
2x2	1728	58.551%	1.157%
3x3	1587	56.818%	0.0%
4x4	1452	56.818%	0.0%
5x5	1323	56.818%	0.0%
6x6	1200	59.091%	0.596%
7x7	1083	63.892%	1.642%
8x8	972	71.42%	1.977%
9x9	867	75.597%	2.46%
10x10	768	81.932%	2.81%
11x11	675	89.574%	1.826%
12x12	588	92.699%	1.887%
13x13	507	95.682%	2.044%
14x14	432	95.739%	1.964%
15x15	363	97.585%	1.732%
16x16	300	96.79%	1.979%
17x17	243	93.409%	2.174%

Table 4: SVM configurations and results using features extracted from equalised RGB image values

4.2 SVM Results

The following tables show the accuracies of the SVM with the accuracy standard deviation, varying colour channels and variance window sizes. This is followed by a plot chart showing the relationship between variance window size and accuracy for the ANN

Window Size	No. Features	Accuracy	Standard Deviation
2x2	576	86.307%	7.451%
3x3	529	61.307%	1.19%
4x4	484	67.642%	1.581%
5x5	441	74.261%	3.097%
6x6	400	79.489%	2.634%
7x7	361	82.244%	3.022%
8x8	324	90.54%	4.777%
9x9	289	92.756%	1.638%
10x10	256	93.58%	1.598%
11x11	225	92.926%	2.13%
12x12	196	93.523%	2.167%
13x13	169	94.148%	2.252%
14x14	144	94.034%	2.19%
15x15	121	93.466%	2.46%
16x16	100	88.75%	2.551%
17x17	81	84.915%	2.78%

Table 3: SVM configurations results using features extracted from equalised greyscale image values

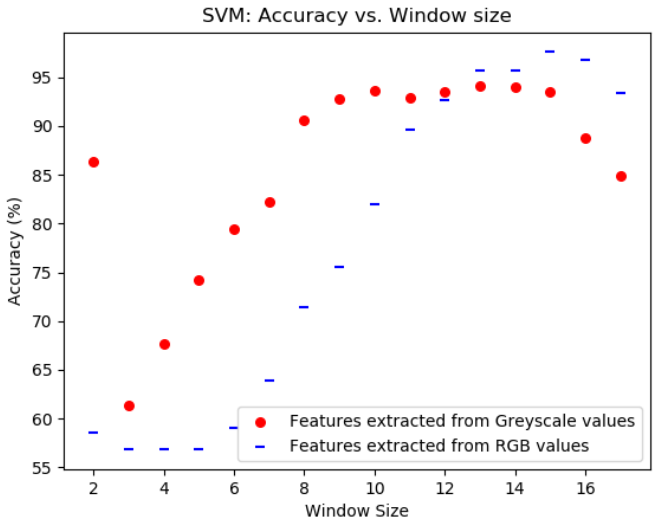


Figure 5: Comparison of the SVM accuracies using features extracted from an RGB image and a greyscale image

There is a strong positive correlation of r=0.662 between the accuracy of the SVM using greyscale extracted features and window size. The correlation between accuracy and window size for the SVM using RGB extracted values is much higher, at r=0.953. These two relationships can be seen in Figure 5. The SVM using greyscale values has a higher average accuracy of 85.62% compared to the other with an average accuracy of 77%, but with p=0.117 for the t test comparing the two, it is not significant enough to draw a conclusion that one outperforms the other.

The result of the SVM using greyscale extracted features and a 2x2 variance window with an accuracy of 86.307% can clearly be

identified as an outlier, as it deviates strongly from the trend as seen in Figure 5. The standard deviation of 7.451% for this particular result is the highest of all results in this study.

4.3 Comparing accuracy of top performing configurations

The two accuracies being compared are the SVM using features extracted from RGB values with a window size of 15x15, and the ANN using features extracted from RGB values with a window size of 8x8. With The ANN mean μ_1=0.96306, the SVM mean μ_2=0.97585, α=0.05, n_1+n_2-2=38 degrees of freedom, S_p=0.01838 and $t_{\alpha/2}$=1.960. Using a t-test gives us a test statistic of T=-2.1448 Since we have |T|=2.1448 > t=1.960, this tells us there is a statistically significant difference in the mean accuracies of the best performing SVM and ANN, with the SVM having the higher accuracy.

4.4 Execution Time of Algorithms

Below are the timings of the most accurate configurations of the two machine learning algorithms tested. The two timings being compared are the SVM using features extracted from RGB values with a window size of 15x15, and the ANN using features extracted from RGB values with a window size of 8x8. With the ANN mean μ_1=0.002593, the SVM mean μ_2=0.005137, α=0.05, $n_1 + n_2 - 2$=38 degrees of freedom, S_p=0.00065619 and $t_{\alpha/2}$=1.960. Using a t-test gives us a test statistic of T=-11.9495. Since we have |T|=11.9495 > t=1.960, there is a statistically significant difference in the mean execution times of the ANN and the SVM classification procedures, with the ANN executing faster.

Algorithm	Accuracy	Classification Time(s)	Standard Deviation	Total Time for 5000 RBCs (s)
ANN	96.306%	0.002593	0.000578	0.07366
SVM	97.585%	0.005137	0.000726	0.14593

Table 5: Comparison of classification times for both algorithms used

5 DISCUSSION

The experiments carried out had the aim of finding a set of preprocessing methods and a machine learning algorithm that could identify red blood cells in a thin blood smear and then classify those blood cells as infected with *Plasmodium falciparum* or healthy, with sufficient accuracy and within a time frame such that it would be considered a viable malaria diagnosis/diagnosis assistance tool. Finding a new solution to diagnosing a major health problem that works faster than previously done could potentially help mitigate the damage malaria causes in many parts of the world.

Previous works have attempted this feat using the blood cell images pixel values as input into various machine learning algorithms. The solution presented attempted to use an abstraction of the pixel values in regions of the blood cell instead of the values of the individual pixels as input features. This is a new approach that was not found in previously published works, described in section 3.6. The idea was that the purple dots where malaria was present in the RBC after the staining process would stand out against the rest of the cell body. This small area of darker colour would cause a higher variation of pixel values in a RBC where malaria was present. With a range of window sizes in which this pixel variation was calculated, the optimal window size could theoretically be found which would most accurately classify blood cells as infected or not. The window size for this method would also depend on the size in pixels of the red blood cell image that has been identified.

In the results presented, the ANN achieved an accuracy greater than 91% for all but one configuration: The use of features extracted from greyscale images with a 17x17 variance window. No distinct pattern could be seen when the results were presented graphically in Fig. 4. One observation that can be made is that the ANN using features extracted from RGB images achieved a higher accuracy than the ANN using greyscale images for all window sizes greater than 4x4. This could be the point at which the use of additional colour information benefits the network more than the use of just pixel intensities provided by the greyscale data. The ANN also achieved a high accuracy from the small window sizes, meaning small levels of abstraction using the pixel variances proved to be enough for the network to generalise a pattern from and classify new data.

The Support Vector Machine achieved accuracy comparable to the ANN for a much smaller range of window sizes. This could be due to the SVM being better at classifying instances of data with less and but more defined features, rather than abstract patterns. This can be seen in Fig. 5 where the more general the feature values became with a higher window size, the better the SVM performed. This trend continues up until a point around a window size of 15, where the input features become too abstract for the SVM to classify them with sufficient accuracy. This accuracy increases sooner with the SVM using greyscale values than with the SVM using RGB value images, as seen in Fig. 5. This may be due to the greyscale images having less features which could be misleading for the SVM during training. The greyscale SVM also started to decrease in accuracy sooner than the RGB SVM, which similarly to the ANN could be from the input features becoming too abstract for the algorithm to classify.

Although an SVM configuration did achieve the highest accuracy among all configurations, the range in which the SVM was sufficiently accurate was much smaller than the ANN. The ANN would be a better candidate for implementing this solution, since it would have the ability to perform in varying conditions, as shown by the results over changing window sizes. Unfocused or slightly blurred images would also work with the ANN, since the variation calculation removes the need for any specific individual deviations in pixel values. General areas of a few pixels changing in value where a parasite is present could be recognised by the algorithm, instead of distinct pixels. The

The two Machine learning algorithms in the proposed solution have configurations that would be considered viable as malaria diagnosis tools. With the ability to classify 5000 blood cells in 0.15 seconds and 0.073 seconds for the SVM and ANN respectively both with accuracy above 96%, these tools meet the WHO standards

described in Section 1.

5.1 Additional findings

It is interesting to note the shape of the trend in Figure 5, where the accuracy increases towards a window size of 10 for greyscale and 15 for RGB features, and then tapers off from there. This may be due to the unknown optimal trade-off between the number of input features and the abstractness of the input features. Where the ANN has the ability to associate weights with certain features values that contribute to a specific result, the SVM may not be able to ignore such values in the same way.

Another observation is the wave-like trend in Figure 4. This could be an avenue for further research, in the trade-off between abstraction and the ability of an ANN to generalise abstract data.

5.2 Limitations of this study

The main limitation of this study was the lack of sufficient sample data. This greatly hindered the reputability of the findings, due to the uniformity of the data used for training and testing both algorithms. With a wider variety of blood cell shapes and different laboratory conditions during image acquisition the algorithms may have performed differently.

A solution for this possible setback was the implementation of histogram equalisation, which negates the possible contrast differences in the images before classification.

The grouping of red blood cells makes individual cell identification a challenge. This solution is ale to identify cells that are alone or close to other bodies but not overlapping. This reduces the number of cells eligible for classification. Overcoming this issue could require more investigation.

6 CONCLUSION

The set of experiments identified multiple possible configurations of a Support Vector Machine and an Artificial Neural Network using pixel window variance values from RGB and greyscale colour channels as input features that would be viable for use as a malaria microscopy diagnosis/ diagnosis assistance tool. The ANN was identified as the more appropriate algorithm should this method be implemented, due to its ability to work with a greater range of input features.

REFERENCES

[1] [n. d.]. ANN Visualisation. https://medium.com/@datamonsters/artificial-neural-networks-for-natural-language-processing-part-1-64ca9ebfa3b2 accessed 2018-05-02.

[2] Adam Blum. 1992. *Neural Networks in C++: An Object-oriented Framework for Building Connectionist Systems.* John Wiley & Sons, Inc., New York, NY, USA.

[3] Dev Kumar Das, Madhumala Ghosh, Mallika Pal, Asok K. Maiti, and Chandan Chakraborty. 2013. Machine learning approach for automated screening of malaria parasite using light microscopic images. *Micron* 45 (2013), 97 – 106. https://doi.org/10.1016/j.micron.2012.11.002

[4] Mohammed E., Mohamed M., Far B., and Naugler. 2014. Peripheral blood smear image analysis: A comprehensive review. *Journal of Pathology Informatics* (2014).

[5] Wijayarathna .G Hirimutugoda .Y.M. 2009. Artificial Intelligence-Based Approach for Determination of Haematalogic Diseases. (2009).

[6] Berend Houwen. 2002. Blood film preparation and staining procedures. *Clinics in Laboratory Medicine* 22, 1 (2002), 1 – 14. https://doi.org/10.1016/S0272-2712(03)00064-7 Interpretation of the Peripheral Blood Film.

[7] Zahoor Jan, Arshad Khan, Muhammad Sajjad, Khan Muhammad, Seungmin Rho, and Irfan Mehmood. 2018. A review on automated diagnosis of malaria parasite in microscopic blood smears images. *Multimedia Tools and Applications* 77, 8 (01 Apr 2018), 9801–9826. https://doi.org/10.1007/s11042-017-4495-2

[8] Walliander M Linder N, Turkki R. 2014. A Malaria Diagnostic Tool Based on Computer Vision Screening and Visualization of Plasmodium falciparum Candidate Areas in Digitized Blood Smears. (2014). https://doi.org/10.1371/journal.pone.0104855.

[9] Kenneth Amiga Kaduki Lucy Gitonga, Daniel Maitethia Memeu. 2014. Determination of Plasmodium Parasite Life Stages and Species in Images of Thin Blood Smears Using Artificial Neural Network. (2014). https://doi.org/10.4236/ojcd.2014.42014

[10] I.G. Maglogiannis. 2007. *Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in EHealth, HCI, Information Retrieval and Pervasive Technologies.* IOS Press.

[11] R Maharaj, J Raman, N Morris, D Moonasar, D N Durrheim, I Seocharan, P Kruger, B Shandukani, and I Kleinschmidt. 2013. Epidemiology of malaria in South Africa: From control to elimination. *SAMJ: South African Medical Journal* 103 (01 2013), 779 – 783.

[12] S. M. Mazalan, N. H. Mahmood, and M. A. A. Razak. 2013. Automated Red Blood Cells Counting in Peripheral Blood Smear Image Using Circular Hough Transform. In *2013 1st International Conference on Artificial Intelligence, Modelling and Simulation.* 320–324. https://doi.org/10.1109/AIMS.2013.59

[13] Roger Boyle DPhil Milan Sonka PhD, Vaclav Hlavac PhD. 1993. *Image Processing, Analysis and Machine Vision* (1 ed.). Springer US.

[14] Emad A Mohammed, Mostafa MA Mohamed, Behrouz H Far, and Christopher Naugler. 2014. Peripheral blood smear image analysis: A comprehensive review. *Journal of pathology informatics* 5 (2014).

[15] World Health Organization. 2010. *Basic Malaria Microscopy Part I. LearnerâĂŹs guide* (2 ed.). World Health Organization. 7–8 pages.

[16] World Health Organization. 2016. *Malaria microscopy quality assurance manual-version 2.* World Health Organization.

[17] World Health Organization. 2016. *World Malaria Report 2015.* World Health Organization.

[18] Nobuyuki Otsu. 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* (1979), 62–66. https://doi.org/10.1109/tsmc.1979.4310076

[19] Sobath Pradeepa Premaratnea, Nadira Dharshani Karunaweerab, Shyam Fernandoc, W Supun, R Pererab, R P Asanga, and S Rajapaksha. [n. d.]. A Neural Network Architecture for Automated Recognition of Intracellular Malaria Parasites in Stained Blood Films. ([n. d.]).

[20] Nicholas E. Ross, Charles J. Pritchard, David M. Rubin, and Adriano G. Dusé. 2006. Automated image processing method for the diagnosis and classification of malaria on thin blood smears. *Medical and Biological Engineering and Computing* 44, 5 (01 May 2006), 427–436. https://doi.org/10.1007/s11517-006-0044-2

[21] Michalski R.S., Carbonell J.G., and Mitchell T.M. 2013. *Machine Learning: An Artificial Intelligence Approach.* Springer Berlin Heidelberg. https://books.google.co.za/books?id=-eqpCAAAQBAJ

[22] Cecilia Di Ruberto, Andrew Dempster, Shahid Khan, and Bill Jarra. 2002. Analysis of infected blood cell images using morphological operators. *Image and Vision Computing* 20, 2 (2002), 133 – 146. https://doi.org/10.1016/S0262-8856(01)00092-0

[23] C. Di Rubeto, A. Dempster, S. Khan, and B. Jarra. 2000. Segmentation of blood images using morphological operators. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, Vol. 3. 397–400 vol.3. https://doi.org/10.1109/ICPR.2000.903568

[24] Christoph Sommer and Daniel W. Gerlich. 2013. Machine learning in cell biology âĂŞ teaching computers to recognize phenotypes. (2013). https://doi.org/10.1242/jcs.123604

[25] Noppadon Tangpukdee, Chatnapa Duangdee, Polrat Wilairatana, and Srivicha Krudsood. 2009. Malaria diagnosis: a brief review. *The Korean journal of parasitology* 47, 2 (2009), 93. https://doi.org/10.3347/kjp.2009.47.2.93
[26] unknown. [n. d.]. Morphological Image Processing. Accessed: 2018-05-01.