diagnostics

MDPI

Article

Deep Learning Based Automatic Malaria Parasite Detection from Blood Smear and Its Smartphone Based Application

K. M. Faizullah Fuhad, Jannat Ferdousey Tuba, Md. Rabiul Ali Sarker, Sifat Momen, Nabeel Mohammed and Tanzilur Rahman *

Department of Electrical & Computer Engineering, North South University, Dhaka 1229, Bangladesh; faizullah.fuad@northsouth.edu (K.M.F.F.); jannat.ferdousey@northsouth.edu (J.F.T.); sarker.rabiul@northsouth.edu (M.R.A.S.); sifat.momen@northsouth.edu (S.M.); nabeel.mohammed@northsouth.edu (N.M.)

* Correspondence: tanzilur.rahman@northsouth.edu

Received: 22 February 2020; Accepted: 15 May 2020; Published: 20 May 2020

Abstract: Malaria is a life-threatening disease that is spread by the Plasmodium parasites. It is detected by trained microscopists who analyze microscopic blood smear images. Modern deep learning techniques may be used to do this analysis automatically. The need for the trained personnel can be greatly reduced with the development of an automatic accurate and efficient model. In this article, we propose an entirely automated Convolutional Neural Network (CNN) based model for the diagnosis of malaria from the microscopic blood smear images. A variety of techniques including knowledge distillation, data augmentation, Autoencoder, feature extraction by a CNN model and classified by Support Vector Machine (SVM) or K-Nearest Neighbors (KNN) are performed under three training procedures named *general training*, *distillation training* and *autoencoder training* to optimize and improve the model accuracy and inference performance. Our deep learning-based model can detect malarial parasites from microscopic images with an accuracy of 99.23% while requiring just over 4600 floating point operations. For practical validation of model efficiency, we have deployed the miniaturized model in different mobile phones and a server-backed web application. Data gathered from these environments show that the model can be used to perform inference under 1 s per sample in both offline (mobile only) and online (web application) mode, thus engendering confidence that such models may be deployed for efficient practical inferential systems.

Keywords: Plasmodium parasites; microscopic; blood smear; data augmentation; CNN; knowledge distillation; Autoencoder; inference performance; floating point operations; deep learning

1. Introduction

Malaria, a life life-threatening disease caused by Plasmodium parasites, is still a severe health concern in large parts of the world especially the third world countries. Almost 219 million cases of malaria, across 87 countries worldwide, were reported by World Health Organization (WHO) in the year of 2017 [1]. WHO designated South-East Asia, Eastern Mediterranean, Western Pacific, and the Americas as high risk zone. Malaria is curable and can be prevented if proper initiatives and approaches are taken effectively, which majorly relies on early diagnosis of malarial parasites [2].

Many techniques have been reported, in the literature, which detects malarial parasites in a patient such as clinical diagnosis [3,4], microscopic diagnosis [5–10], rapid diagnostic test (RDT) [11–13] and polymerase chain reaction (PCR) [14–16]. Conventional diagnostic methods like clinical diagnosis and PCR are performed in laboratory settings, and their efficiency and accuracy largely depend on the level of available human expertise [17]. Such expertise is inadequately present in unreached remote areas where malaria can be predominant.

RDT and microscopic diagnosis are two of the most impactful malaria diagnosis methods which make a very large contribution to malaria control today [13]. RDT is effective diagnosis tool as it does not require any trained professional or microscope and can provide diagnosis within 15 min [18]. However, according to WHO [19] and others [20,21] RDT has few shortcomings that includes lack of sensitivity, inability to quantify parasite density and differentiate among *P. vivax*, *P. ovale* and *P. malariae*, higher cost compared to light microscope, and susceptibility to damage by heat and humidity. Microscopic systems do not suffer from these shortcomings and is considered to be effective for malaria parasite detection [13,22], however this technique requires the presence of a trained microscopists [23]. Automatic microscopic malaria parasite detection, which involves the acquisition of the microscopic blood smear image (for example by smartphone as demonstrated in [24,25]), segmentation of the cells and classification of the infected cells, can be an effective diagnostic tool [26]. It is to be noted that successful segmentation of blood cells and identification of malaria parasites can be used in conjunction to perform counting. Cell segmentation is a well researched area and good performance has already been reported in multiple studies [27–30]. Previous works that concentrated on the classification of infected cells involve tools and techniques from image processing [31–33], computer vision [34–36] and machine learning [37–39]. However, solutions for classification of infected cells, which are both accurate and computationally efficient, have not been studied to the best of our knowledge. As an example, [40] reports an accuracy of 99.52% but the proposed model uses over 19.6 billion floating point operations (flops). This precludes the model from being useful in power constrained device.

In this study we trained multiple accurate and computationally efficient models for malaria parasite detection in single cells using a publicly available malaria dataset [24]. Our contributions reported in this article are mainly three-folds:

- While doing experiments on the malaria dataset, we found that certain samples were mislabeled. These mislabeled samples were corrected and the corrections can be found in [41] for future research.
- Unlike previous works carried out for malaria parasite detection, our trained models are not only highly accurate (99.23%) but also the order of magnitude is more computationally efficient (4600 flops only) compared to previously published work [40].
- For better understand the performance of the model in low resources settings, it was deployed for inference in multiple mobile devices as well as a web application. We find that in such situation the model can be used to find accurate per cell classification prediction within 1 s.

It is conceivable that these contributions can play a significant role towards building a fully automated system for malaria parasite detection in the future.

2. Relevant Work

Malaria being a life threatening disease has caused deep research interests among the scientists all over the world. Earlier, malaria was mostly diagnosed in the laboratory setting requiring a great deal of human expertise. Automatic systems such as those relying on machine learning techniques were initially studied to overcome this problem. Techniques reported in this domain of study mostly considered the hand-crafted features in decision making. For example, [42,43] relied on morphological factors for feature extraction and applied SVM and Principal Component Analysis (PCA) [44] for the classification purpose. However, the accuracy achieved through these kinds of model is low compared to the more recently studied deep learning based techniques.

For automatic detection of malaria pathogens from the microscopic images, Convolutional Neural Network (CNN) [45] received much attention from the researchers in recent times. Dong et al. for example evaluated the performances of three popular Convolutional Neural Network [46], namely LeNet-5 [47], AlexNet [48] and GoogLeNet [49]. In addition, they trained an SVM classifier for comparison purposes and concluded that CNN is advantageous over SVM in terms of the capacity of learning image features automatically. To find the optimal layer of a pre-trained model to extract features from underlying malaria parasite data, Rajaraman et al. evaluated the performances of AlexNet, VGG-16 [50], ResNet-50 [51], Xception [52], DenseNet-121 [53] along with their custom-built model [24]. Liang et al. reported 97.37% accuracy in detection with the help of their 16-layered CNN model and claimed that their model is superior to transfer learning models [54]. Hung et al. pre-trained a model on Imagenet [55] but fine-tuned it on own data for detecting malaria parasite [35].

Most of the reported studies in general focused on improving the accuracy using traditional machine learning or deep learning based techniques. A few notable exceptions are the works of Quinn et al. [25] and Rosado et al. [56] who also tackled the problem of computational efficiency of their models. However, both studies reported a notable drop in model accuracy in pursuit of computation efficiency. Rosado et al. also explored the use of smartphones to detect malaria parasites and white blood cells. They reported sensitivity and specificity of 80.5% and 93.8% for trophozoite detection and 98.2% and 72.1% respectively for white blood cells using Support Vector Machine (SVM) [57]. Using the same classifier, they achieved a sensitivity and specificity of 98.2% and 72.1% respectively for white blood cells. The use of deep learning technique, particularly those that leverage transfer learning [31,40,58] yield superior results when considering classification metrics. Unfortunately, the models proposed in those studies are also quite expensive in terms of the required computational resources. Also notable is the fact that while Rosado et al. showed that their model can be deployed on mobile phones, their demonstration did not include low cost mobile phones commonly used in the poorer regions of the world.

In contrast, in this work, we propose several deep learning models which achieve classification performance comparable to the previously reported highly accurate deep learning based solutions. In addition, our models are efficient in terms of required computational resources and have been demonstrated to work efficiently on smart mobile devices, including that are available at very low cost.

3. Methodology

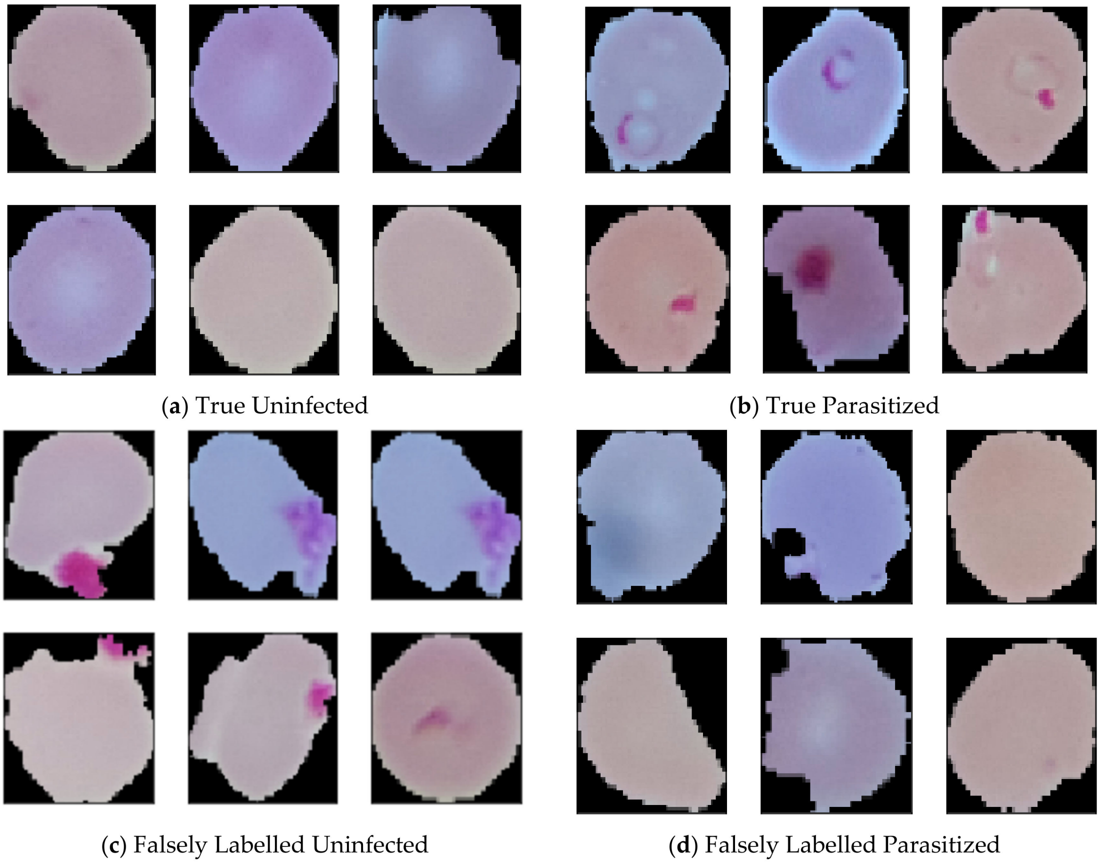
In order to conduct a series of experiments, publicly available malaria dataset was used. Data collection and data preprocessing techniques are discussed in the following subsequent subsections. Out of the series of experiments, we choose our best model in terms of both performances and effectiveness thereby, which is discussed in proposed model architecture subsections. Experimental details and experimental settings are discussed in training details subsections. Training of the models is discussed under three training procedure which are *general training* procedure, *distillation training* procedure and *autoencoder training* procedure, details are provided in the designated subsections.

3.1. Data-Set

Malaria dataset contains 27,558 cell images classified into two groups called parasitized and uninfected cells, where each cell contains an equal number of instances. Data was taken from 150 P. falciparum and 50 healthy patients and it was photographed at Chittagong Medical College Hospital, Bangladesh using a smartphone by placing it on the conventional light microscope [26]. Manual annotation was performed later by an expert slide reader at the Mahidol-Oxford Tropical Medicine Research Unit. In this data, parasitized samples mean that there is the presence of Plasmodium, whereas the uninfected samples refer to the absence of Plasmodium but there may be presence of other objects like staining artefacts/impurities.

While studying the dataset, some of the labelled data raised suspicion of whether they were correctly labelled. Some of the data seems like uninfected but labelled as parasitized, where some parasitized images are labelled as uninfected. To confirm this rising issue we consulted with an expert. The expert confirmed that some of the data are genuinely mislabeled which was later manually annotated as per the presence and absence of malarial parasites. While annotating, suspicious, and falsely labelled data was put aside, which resulted in the reduction of data from 27,558 to

26,161. After removing 647 falsely labeled and suspicious parasitized data, the amount of current parasitized data stands 13,132. In this article, correct parasitized data is considered as true parasitized, and suspicious data is considered as false parasitized. For uninfected malaria data, 750 suspicious images was found, which was named as false uninfected. After keeping away those data, total true uninfected data stands 13,029. Some data from the dataset are depicted in Figure 1.



(**a**) True Uninfected

(**b**) True Parasitized

(**c**) Falsely Labelled Uninfected

(**d**) Falsely Labelled Parasitized

Figure 1. (**a**) Correctly labelled uninfected images (**b**) Correctly labelled parasitized images (**c**) Falsely labelled uninfected images and (**d**) Falsely labelled parasitized images.

3.2. Data Preprocessing

In supervised learning, the behaviour and performances of the model entirely depends on the data that is fed. Therefore, data preprocessing plays a vital role towards conducting experiments. Considering that, in this work manually corrected images were resized as per the model input, and image patches were rescaled to map the features within 0 to 1 range which led to getting faster convergence. Data augmentation as per Table 1 was also applied to training data preserving the semantic meaning, which helps to improve the model performances. Figure 2 depicts resizing augmented images.

Table 1. Augmentation table.

Augmentation Type	**Parameters**
Random Rotation	20 Degree
Random Zoom	0.05
Width Shift	(0.05, −0.05)
Height Shift	(0.05, −0.05)
Shear Intensity	0.05
Horizontal Flip	True

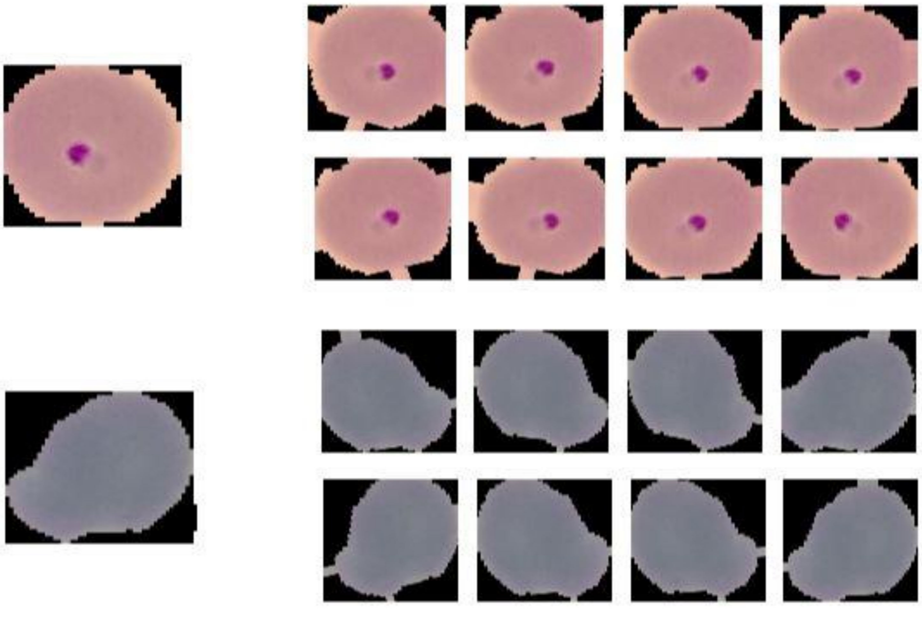


Figure 2. Image before resizing and performing augmentation is depicted in (**a**) and image after resizing and applying augmentation is depicted in (**b**).

3.3. Proposed Model Architecture

To serve the purpose of detecting malaria parasite from blood smear (exactly the similar kind of blood smear collected by [24]), in this article, an autoencoder-based architecture is proposed, which is shown in Figure 3. Autoencoder [59] is a specific type of artificial neural network which compresses input data into lower-dimensional latent space representation and finally reconstruct output from this representation shown in Equations (1) and (2).

$$\overline{X_i} = Decoder(Encoder(X_i)) \tag{1}$$

$$L_\theta = C_\theta\left(\overline{X_i},\ X_i\right) \tag{2}$$

Here, Mean Square Error (MSE) loss function C calculates the loss between i number of the original and reconstructed image.

The primary purpose of an Autoencoder is dimensional reduction [60]. However, in this task, it is used as a classifier, inspired by [61,62]. Autoencoder is composed of two main components: Encoder and Decoder. For the classification task, the decoder is replaced by the fully connected layer which allowed Autoencoder to classify the expected classes. The complete process will be discussed in the subsequent section.

Encoder: Encoder compresses the input to latent space representation with the least possible distortion. For $X_n = x_1, x_2, x_3, \ldots, x_n$ set of input images belongs to training set, encoder compress it to $K_n = k_1, k_2, k_3, \ldots, k_n$ where K_n is the set of latent representation of X_n.

$$k_1, k_2, k_3 \ldots\ldots k_n = Encoder(x_1,\ x_2,\ x_3 \ldots\ldots x_n) \tag{3}$$

$$K_n = Encoder\ (X_n) \tag{4}$$

The proposed encoder is composed of three Convolutional layers where a Max-pooling layer follows each layer. For performing the convolutional operation in each layer, the kernel size is defined as (3 × 3) with the same padding and 1-pixel stride. The kernel number for the first convolutional layer was set 16 where second and third layers were respectively 8 and 4. ReLU activation function, shown in Equation (5), was applied in encoder's hidden units to introduce non-linearity to the neuron's output.

$$S = \max(0, M) \tag{5}$$

Here, S is the output after applying non linearity on matrix M.

To sample down the features map, max-pooling layer shown in Equation-6 is applied with window size 2 × 2 and strides 1.

$$Z = max_{i,j=1}^{h,w}\left(M_{i,j}\right) \tag{6}$$

Z is the output matrix containing maximum value of each patch from input matrix M.

Decoder: Decoder reconstructs the image R_n = r_1, r_2, r_3, . . . , r_n from the latent representation K_n = k_1, k_2, k_3, . . . , k_n is shown in Equations (7) and (8).

$$r_1\ , r_2\ , r_3 \ldots\ldots r_n = Decoder\ (k_1\ , k_2\ , k_3 \ldots\ldots k_n) \tag{7}$$

$$R_n = Decoder\ (K_n) \tag{8}$$

The decoder consists of 4 Deconvolutional layers and three Up-sampling layers. The kernel size for all the Deconvolutional layers is 3 × 3 with strides size 1 having the same padding and number of kernels were defined as 4, 8, 16 and 3 respectively. Deconvolutional layer is the opposite of convolutional layer and unlike Convolutional layer, instead of mapping 3 × 3 features into 1 pixel, Deconvolutional layers map 1 pixel to 3 × 3 features vector. ReLU activation function was applied to the hidden units to introduce non-linearity. Up-sampling with window size 2 × 2 was applied to get closer input image to reconstruct it from the latent representation.

While testing, decoder is replaced by a flatten and two fully connected layers to serve our purpose of detecting malaria parasite from a blood smear. Hence, the final model is composed of 9 layers among which three are Convolutional layers, three are Max-pooling layer, one Flatten layer and two fully connected layers. Finally, fully connected layer is followed by a softmax layer to distribute predicted probabilities over two classes according to the objective of this task. The outline of our proposed model architecture is given in Figure 3.

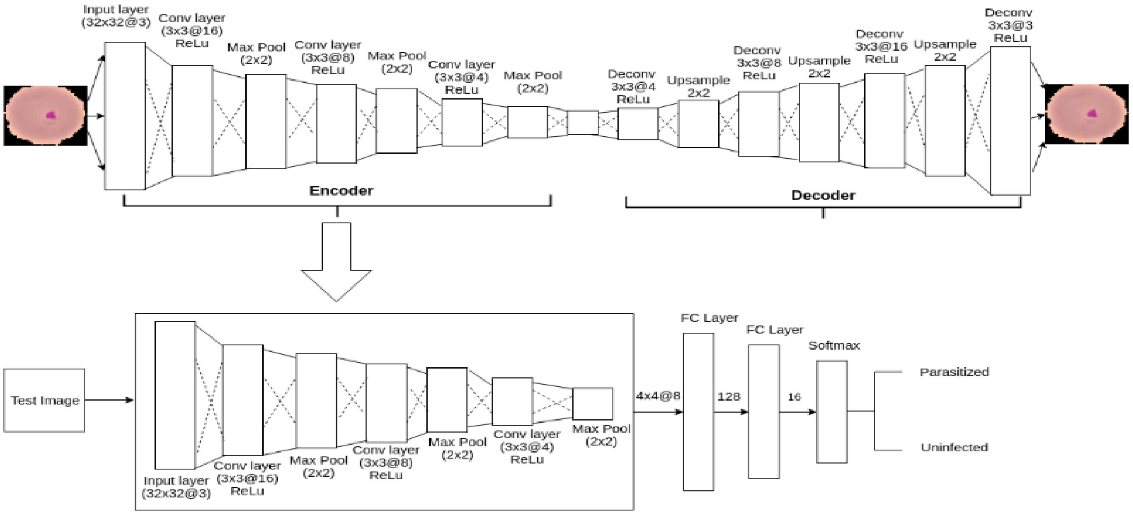


Figure 3. The outline of autoencoder model.

3.4. *Training Details*

To find the optimum and efficient model, different approaches was investigated. Among the series of experiments, feature extractions and classification using Support Vector Machine and K-Nearest

Neighbors (KNN) along with augmentation and without augmentation is discussed under General training procedure. In Distillation training procedure, model pruning and knowledge transformation from teacher to student model is discussed. The third procedure is the Autoencoder training procedure where training and testing procedure of an Autoencoder based model is discussed. For general training and knowledge distillation training, a custom 8-layered Convolutional Neural Network model, shown in Figure 4, was used as a feature extractor. In all the training procedure, total data was divided into training, testing and validation group as per the ration 80:10:10. All experiments was performed on machine Ubuntu 16.04, system with Intel@ Core i5-6500 CPU @ 3.20 GHz processor, 8 GB RAM, 1 TB HDD, Python @3.6.7, Keras @2.2.4 having TensorFlow @1.13.0 backend.

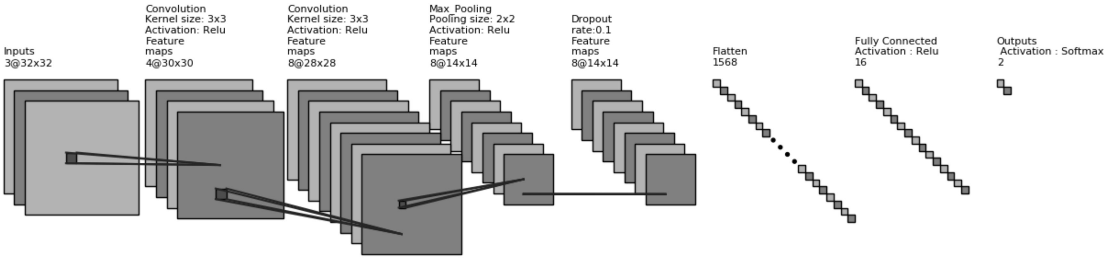


Figure 4. Custom 8 layers model architectures which is taken as a feature extractor for the general training procedure and knowledge distillation training procedure.

3.4.1. General Training

In order to optimize and to obtain better performances, data augmentation, without augmentation, classification using SVM and KNN was performed. For the augmentation experiment, augmentation as per the Table 1 was applied first. While performing 64 × 64 augmentation experiment, original image was resized to (64, 64) before applying augmentation where augmentation was performed after resizing original image into (32, 32) for the experiment of 32 × 32 augmentation. In both experiments, data was fed into the model according to batch size 32. For the without augmentation training, plain data was fed into the model according to batch size 64 after resizing the original image to (64, 64) or (32, 32) as per the experiment requirement. Weights and biases of all trainable parameters were randomly initialized at the beginning of training. To optimize the weights and biases, Stochastic Gradient Descent (SGD) [63] optimizer was used while performing the forward propagation. The learning rate was set to 0.001 with a momentum of 0.9. While performing back-propagation to calculate the error and update the weights and biases, cross entropy, shown in Equation (9), was used as a loss function. Predicted result and combined validation loss and accuracy of augmentation and without augmentation experiment for both image sizes are depicted in Figure 5; Figure 6, respectively.

$$Loss = -(y\,log(p) + (1-y)\log(1-p)) \tag{9}$$

In the experiment of CNN-SVM and CNN-KNN, a CNN model was used to extract the features. Later these features were classified through SVM or KNN as per the objective of our experiment. The outline of this architecture is depicted in Figure 7.

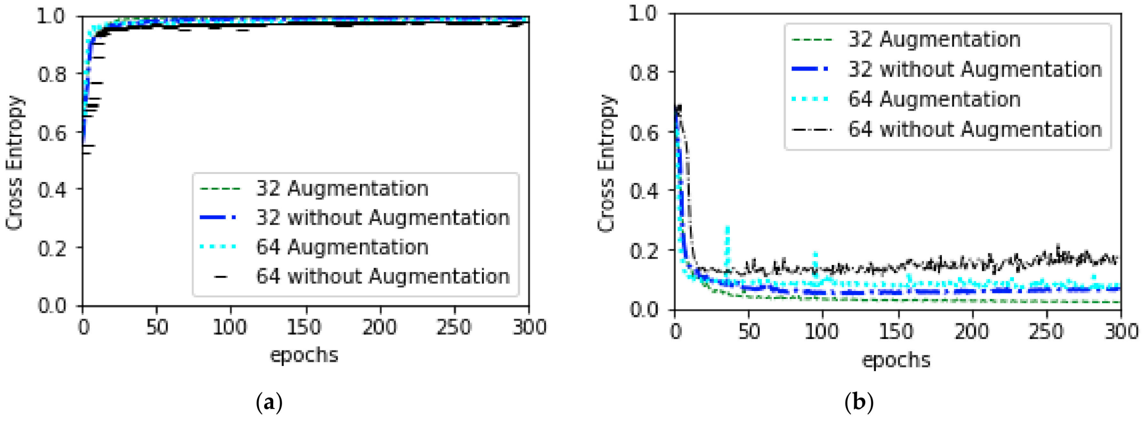


Figure 5. Combine validation accuracy and validation loss: (**a**) depicts combine validation accuracy and (**b**) depicts combine validation loss.

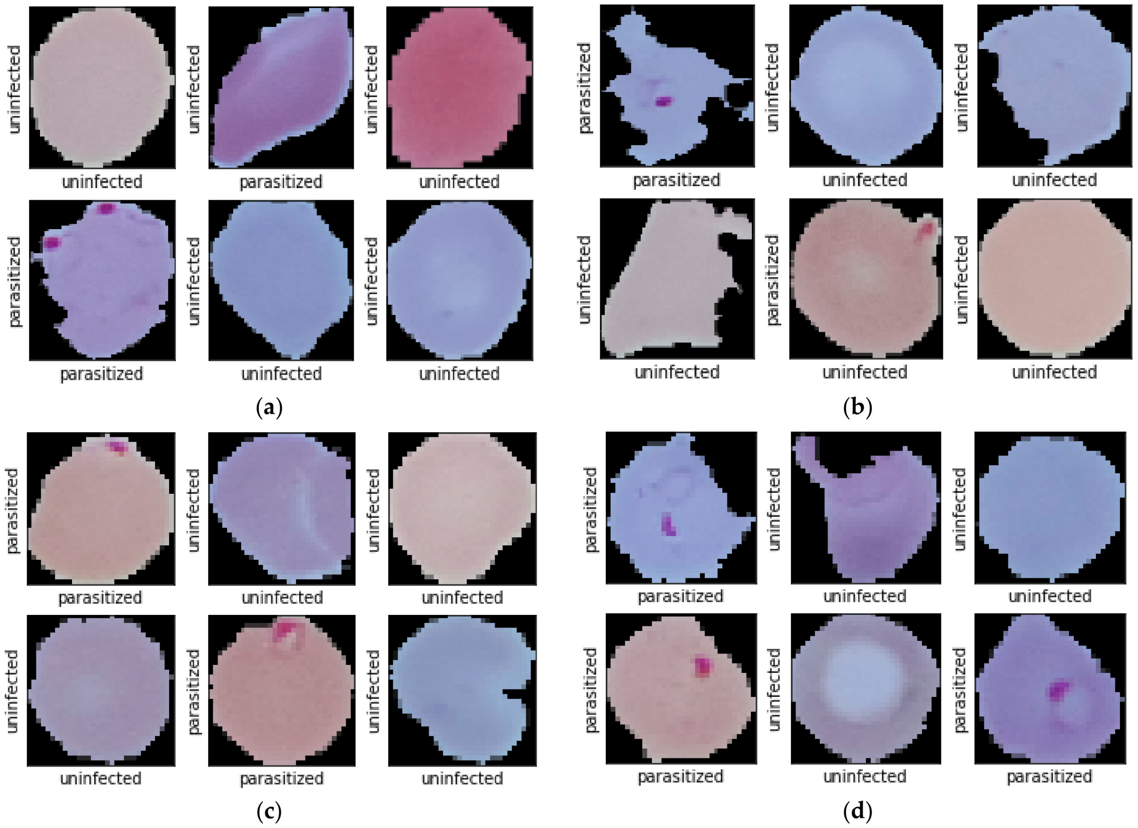


Figure 6. Predicted results where X-axis represents ground-truth value and Y-axis represents predictions of the model: (**a**) depicts predicted result of image 64 × 64 with Augmentation (**b**) depicts predicted result of image 64 × 64 without Augmentation (**c**) depicts predicted result of image 32 × 32 with Augmentation and (**d**) depicts predicted result of image 32 × 32 without Augmentation.

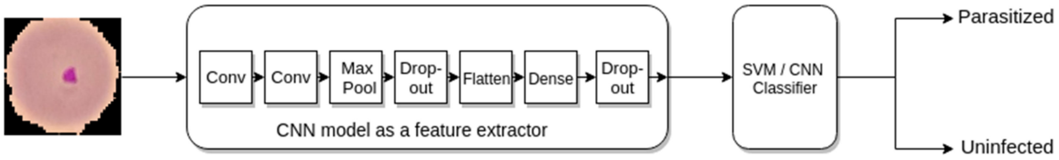


Figure 7. Modified CNN-KNN or CNN-SVM architecture.

Support vector machine (SVM) and K-Nearest Neighbor (KNN) are the core machine learning algorithms that were used as a classifier by implicitly mapping their inputs into high-dimensional feature spaces [57,64]. Earlier work [65–67] reported the effectiveness of SVM or KNN classifier on the Convolutional neural network's extracted features instead of using the softmax layer. Being motivated with this work, we trained SVM and KNN models after extracting features from a CNN model (depicted in Figure 4) as shown in Equations (10) and (11).

$$Label_{\propto}^{pred} = S\Big(C^{l-1}(X_{\propto})\Big) \tag{10}$$

$$Label_{\propto}^{pred} = K\Big(C^{l-1}(X_{\propto})\Big) \tag{11}$$

Here, X is the images of training set, C is the feature extractor model, l is the number of layer and S, K is respectively SVM and KNN classifier.

While training CNN-SVM or CNN-KNN model, to get better performance, hyperparameters were fine-tuned. In CNN-SVM model 98.93% accuracy was achieved by using 'RBF' kernel and setting gamma and regularization parameter value to 0.001 and 10 respectively. The best accuracy of CNN-KNN model was 99.12% when 3 was defined as neighbor's number. Model performances under general training procedure are shown in Table 2.

Table 2. Model performances after general training.

Image Size	Augmentation	Epochs	Validation Accuracy	Validation Loss
(32,32)	No	300	0.9982	0.06
(64,64)	No	300	0.9748	0.16
(32,32)	Yes	300	0.9920	0.02
(64,64)	Yes	300	0.9813	0.08

3.4.2. Distillation Training

As a part of the experiment, model pruning method like knowledge distillation was also explored. Knowledge distillation is a model compression method shown in Equation (12) where knowledge is transferred from a larger model (teacher model) to a smaller model (student model). In this method, a smaller model is trained to learn the exact behavior of a complex and bigger model. This method is proposed by Bucila [68] and generalized by Hinton [69].

$$Z_i = \frac{\exp\left(\frac{k_i}{T}\right)}{\sum_j \exp\left(\frac{k_i}{T}\right)} \tag{12}$$

In our experiment, a pretrained 146 layers custom ResNet model [70] was taken as a teacher model, whereas a custom 8 layers CNN model (showed in Figure 4) acted as a student model. In order to mimic the teacher model, initially using the pretrained teacher model, the softmax value of training images was recorded and also the training label was predicted. Later, the softmax value of training images was swapped if the teacher model mis-predicts the training label. This corrected softmax value was used as a training label while training student model, shown in Equations (13) and (14).

$$\gamma_i^{\tau} = T_m(X_i) \tag{13}$$

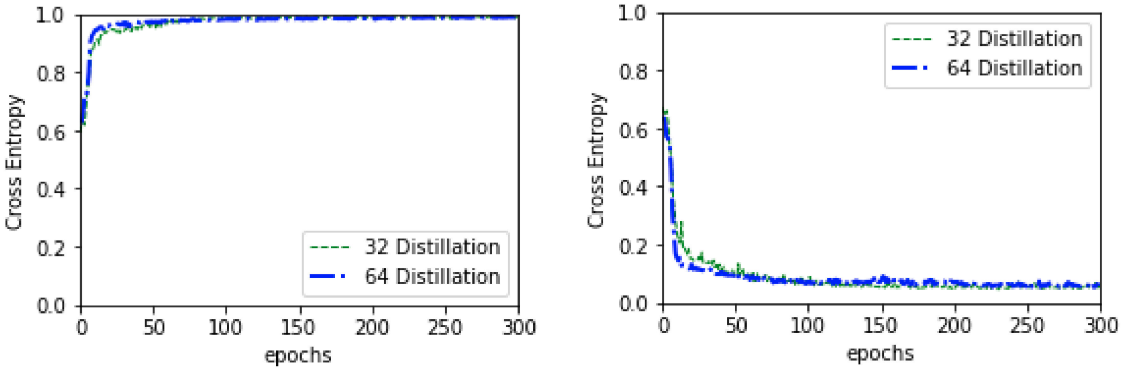
$$\gamma_i^{S_{train}} = \begin{cases} \gamma_i^{\tau}, & if\ argmax\left(\gamma_i^{\tau}\right) == argmax\left(\gamma_i^{\rho}\right) \\ \gamma_i^{\rho}, & Otherwise \end{cases} \tag{14}$$

Here, $\gamma_i^{S_{train}}$ is training label of student model
γ_i^{τ} is the softmax prediction of teacher model

γ_i^ρ is the actual label

T_m is the Teacher model

Stochastic Gradient Descent (SGD) was used as an optimizer with initial learning rate of 0.01 and momentum of 0.5 while training the student model. To update the weight, a decay of 1e-6 was applied using Keras learning rate decay function. To calculate the loss and update the weights and biases of all trainable parameters Cross Entropy Loss function shown in Equation (9) was applied. Combine validation accuracy and loss is shown in Figure 8, and the predicted result of this model are shown in (Figure S1).



(**a**) (**b**)

Figure 8. Combine validation accuracy and validation loss of knowledge distillation training: (**a**) depicts combine validation accuracy and (**b**) depicts combine validation loss.

After training model performance are depicted in Table 3.

Table 3. Model performances after performing Knowledge Distillation.

Image Size	Augmentation	Epochs	Validation Accuracy	Validation Loss
(32,32)	Yes	500	0.9897	0.04
(64,64)	Yes	500	0.9859	0.05

3.4.3. Autoencoder Training

Finally, autoencoder based architecture was explored to increase the model's performance. It is found that the autoencoder based experiments outperforms other experiments. Autoencoder is composed of two main components which are Encoder and Decoder discussed in the proposed model subsection. The main aim of an Autoencoder is dimensionality reduction, but to detect malaria parasite, it is used as a classifier. This classifier can classify infected and parasitized images from a microscopic blood smear. To train the classifier, at first, full Autoencoder network was trained with the augmented training images where augmentation was performed as per Table 1. Mean Square Error (MSE) loss function shown in Equation (15) was used while training to calculate the losses. To minimize the loss and update the weights RMSprop [71] optimizer was used with 0.001 learning rate. Figure 9 depicts the few reconstructed images after training Autoencoder 500 epochs.

$$Loss_{MSE} = \frac{1}{k}\sum_{j=1}^{k}(X_i - \overline{X}_i)^2 \tag{15}$$

After training Autoencoder network, Decoder was replaced by a fully connected layer where a softmax layer follows the fully connected layer in order to distribute predicted probabilities into uninfected and parasitized class. This modified network was trained for 300 epochs after freezing encoder layers which helps fully connected layers to learn the image features without changing encoder's weights and biases. Cross Entropy loss functions was used to calculate the losses. SGD optimizer with learning rate 0.001 and momentum 0.9 was used to optimize and update the weights and biases. Figure 10 depicts the training behavior of this modified model.

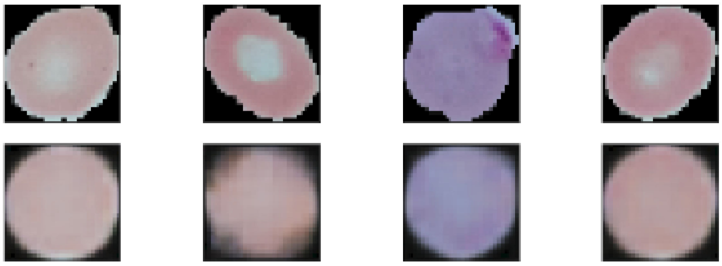


Figure 9. Some reconstructed images after training Autoencoder. Top images are the original image and bottom images are reconstructed images.

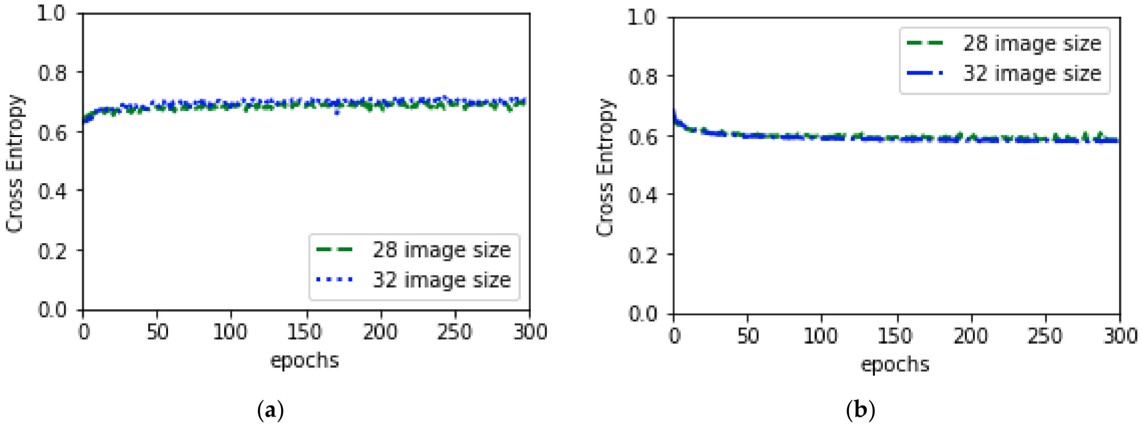


Figure 10. Training behavior of Autoencoder after freezing encoder layer: (**a**) depicts the validation accuracy for image (32, 32) and (28, 28) and (**b**) depicts validation loss for image (32, 32) and (28, 28) while training Autoencoder 300 epochs.

Further, Autoencoder model was trained after unfreezing the encoder layer, which led to achieving better performance by the model. While training, the hyper-parameter was set as like before. The experiment was done for both 32 × 32 and 28 × 28 images. The final behavior of the model in both experiments is depicted in Figure 11 and Table 4.

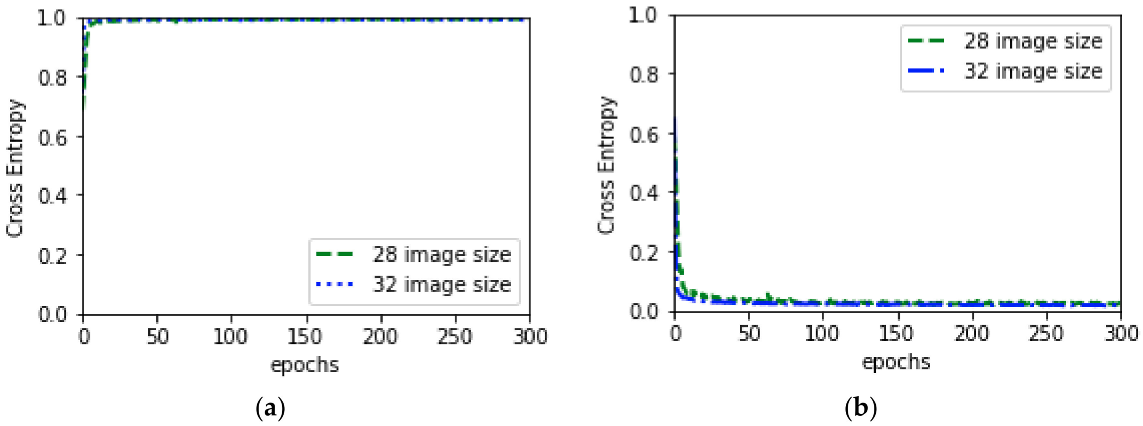


Figure 11. Final behavior of Autoencoder: (**a**) depicts the combine validation accuracy and (**b**) depicts combine validation loss while training Autoencoder 500 epochs.

Table 4. Model performances after training Autoencoder 500 epoch.

Image Size	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
(28,28)	0.9943	0.018	0.9924	0.025
(32,32)	0.9941	0.019	0.9900	0.032

4. Result Analysis

To develop an efficient and highly accurate model for the detection of the malaria parasite from segmented cell images, a series of experiments involving both machine learning and deep learning techniques were performed. The models were evaluated using different performance metrics such as Test Accuracy (Test Acc), F1 Score, Precision (Precs.), Sensitivity (Sens.) and Specificity (Spec.). It is important to note that the size of the model was also considered as an important factor along with performance to ensure the viability of the models in lower-cost smartphones. The performance of our models explained by the above-mentioned evaluation metrics can be found in Table 5.

Table 5. Model performances on different experiment.

Image Size	Aug	Method	Test Acc	Test Loss	F1 Score	Precs.	Sens.	Spec.	Size (KB)
(32,32)	Yes	-	0.9915	0.03	0.9914	0.9861	0.9960	0.9865	233.60
(32,32)	No	-	0.9877	0.05	0.9876	0.9892	0.9861	0.9893	233.60
(64,64)	Yes	-	0.9843	0.07	0.9839	0.9836	0.9840	0.9842	954.50
(64,64)	No	-	0.9755	0.15	0.9751	0.9851	0.9650	0.9855	954.60
(32,32)	Yes	Distillation	0.9900	0.04	0.9900	0.9877	0.9920	0.9878	233.60
(64,64)	Yes	Distillation	0.9885	0.04	0.9882	0.9929	0.9836	0.9932	954.60
(28,28)	Yes	Autoencoder	0.9950	0.01	0.9951	0.9929	0.9880	0.9917	73.70
(32,3)	Yes	Autoencoder	0.9923	0.02	0.9922	0.9892	0.9952	0.9917	73.70
(32,32)	Yes	CNN-SVM	0.9893	-	0.9918	0.9921	0.9916	-	-
(32,32)	Yes	CNN-KNN	0.9912	-	0.9928	0.9911	0.9923	-	-

Table 5 depicts the performance of the models on the test data as obtained from the experiments. In terms of size and model performance, models trained using distillation, and autoencoder is compatible to serve the purpose of being computationally efficient while retaining high classification accuracy. As the autoencoder trained model is smaller in size (only 73.70 KB) and outperforms models obtained from distillation and other techniques, these models may be considered for practical validation on smartphones. The two autoencoder based models differ in their input image resolution. One uses 28 × 28 images and the other accepts 32 × 32 images. Figure 12 and (Figure S2) shows the prediction of autoencoder model experimented on 32 × 32 images and the image resolution of the two respective autoencoder based models. Although the model that accepts 28 × 28 images has slightly higher accuracy, we preferred to use the model which uses the slightly higher resolution image for deployment in mobile phones and the web-application.

Autoencoder model experimented on 32 × 32 falsely predicts 23 images while testing on 3000 test images. The confusion matrix is placed in Figure 13 to show the model performances on test data.

According to Figure 13, it is clear that among the 23 falsely classified images, 16 are false positive, and 7 are false negative. Some of the misclassified images are shown in (Figure S3). Studying upon all those images, we found that, some of the images are hard to classify even for the human eyes. So, with the special consideration of this issue, we can say that autoencoder based model performs well on malaria data.

In addition, the generalizability of the model was established by evaluating its performance on images from a separate publicly available dataset [72]. The comparable performance achieved with this dataset further confirm that the proposed model is not localized or biased to the dataset on which it was trained. Experimental details and results on this new dataset can be found in the Appendix A.

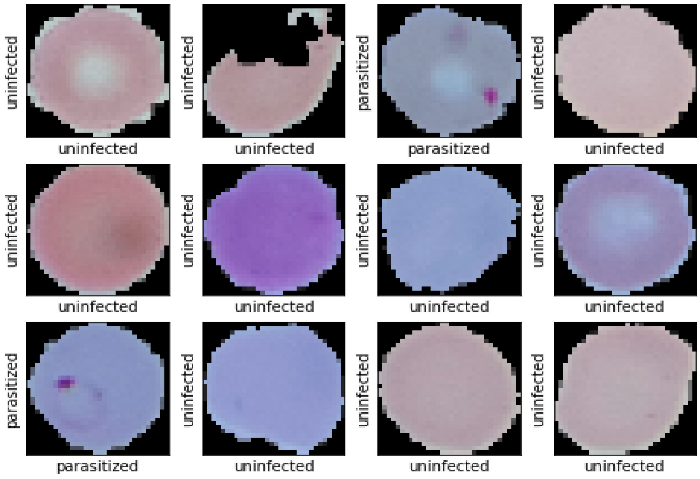


Figure 12. Some predicted result of Autoencoder based model. Here X-axis represents Ground Truth value and Y-axis represents model prediction.

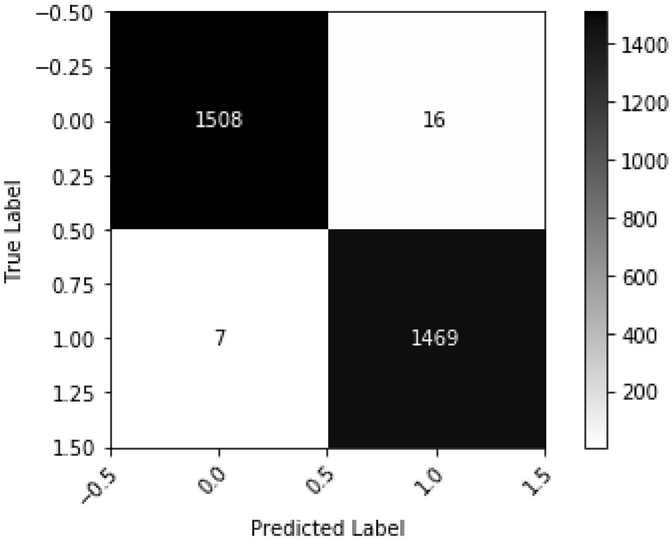


Figure 13. Confusion Matrix of Autoencoder based model based on test data.

5. Model Deployment

To demonstrate the robustness and compatibility of the developed model, both web-based and mobile-based applications were developed. This development clearly shows that this model can be beneficial for the automatic diagnosis of Malaria in a restricted resource environment.

5.1. Mobile Based Application

Mobile phone has become ubiquitous these days and is an essential part of our day to day life activities. It is not only used for communication but also uses in many other purposes. Mass people use this powerful device even in a developing country due to being available at a reasonable cost. Keeping those in mind, the developed model was integrated with a smartphone application to demonstrate that Smartphone's based automatic detection of the malaria parasite is quite possible. The model can work independently in the mobile app without needing internet connection and can help an individual without any technical expertise to detect malaria parasite from the blood smear. Figure 14 depicts the pictorial diagram of a mobile-based solution.

To deploy a model into the smartphone, the model needs to be converted into TensorFlow Lite model as it is the best solution to execute the trained model accurately in limited memory and computational power. Along with this, TensorFlow Lite converter provides options that allow to reduce file size and increase the speed of execution, with some trade-offs. Considering all those, for making mobile-based application at first, trained model was converted into TensorFlow Lite. After that, an interpreter was initialized for loading the model. An interpreter is a program that converts programs written in a high-level language into machine code understood by the computer. The model was then initialized after it was appropriately loaded. To process image, the mobile application provides the option for capturing image from the storage. After selecting the image from the gallery or taking a picture from the camera resizing of the image into 32 × 32 is performed as per model input. Later, it is passed to the CNN model as an input for invoking the interpreter to get the result. Then the model can run and analyze the image. After the analysis, the result would be given by the model in binary, which would represent whether the result is parasitized or uninfected.

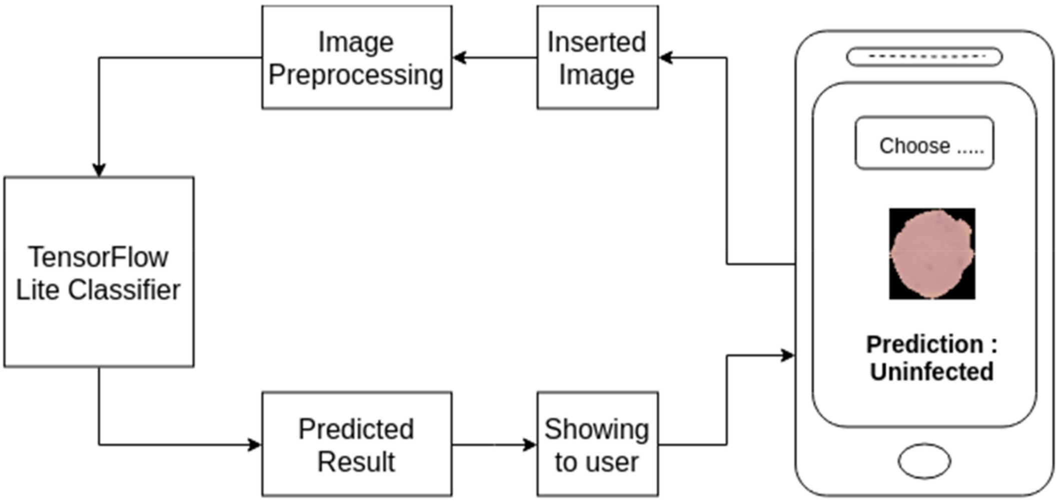


Figure 14. Outline of mobile based service.

The validity of this developed model was tested on different Smartphone. Table 6 shows the detailed information of testing result on the different operating system and its resources specification.

Table 6. Compatibility of developed model in different devices.

Version	Codename	API	Distribution	Device	Model	Ram	Comments
2.3.3–2.3.7	Gingerbread	10	0.30%	-	-	-	Almost obsolete OS
4.0.3–4.0.4	Ice Cream Sandwich	15	0.30%	-	-	-	Almost obsolete OS
4.1.X	Jelly Bean	16	1.20%	-	-	-	Almost obsolete OS
4.4	Kitkat	19	6.90%	Samsung	GT-190601	2 GB	Work Perfectly
5	Lollipop	21	3.00%	Google	Nexus 4 (Emulator)	2 GB	Work Perfectly
6	Marshmallow	23	16.90%	Google	Nexus 5 (Emulator)	1 GB	Work Perfectly
7	Nougat	24	11.40%	Xiaomi	Readmi Note 4	3 GB	Work Perfectly
8	Oreo	26	12.90%	Samsung	S8 (Emulator)	2 GB	Work Perfectly
8.1	-	27	15.40%	LG	LM-X415L	2 GB	Work Perfectly but latency is higher compared to other devices.
9	Pie	28	10.40%	Xiaomi	A2 Lite	4 GB	Work Perfectly

The graphical interface of developed mobile based application is shown is Figure 15.

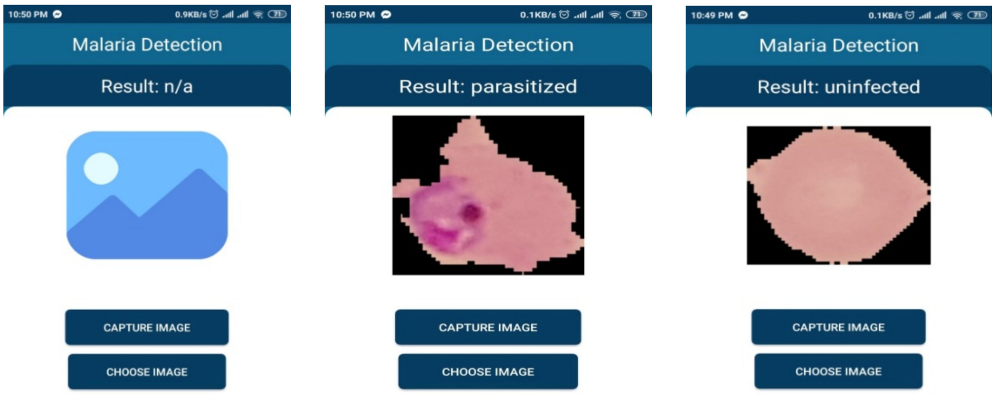


Figure 15. Model behavior on smartphone-based service.

5.2. Web Based Application

A web-based application was developed to demonstrate the model compatibility on server. This application was developed using Flask. As it is a part of research work, service was kept in local machine apart from deploying in online. Figure 16 depicts the behavior of this model in the web-based application.

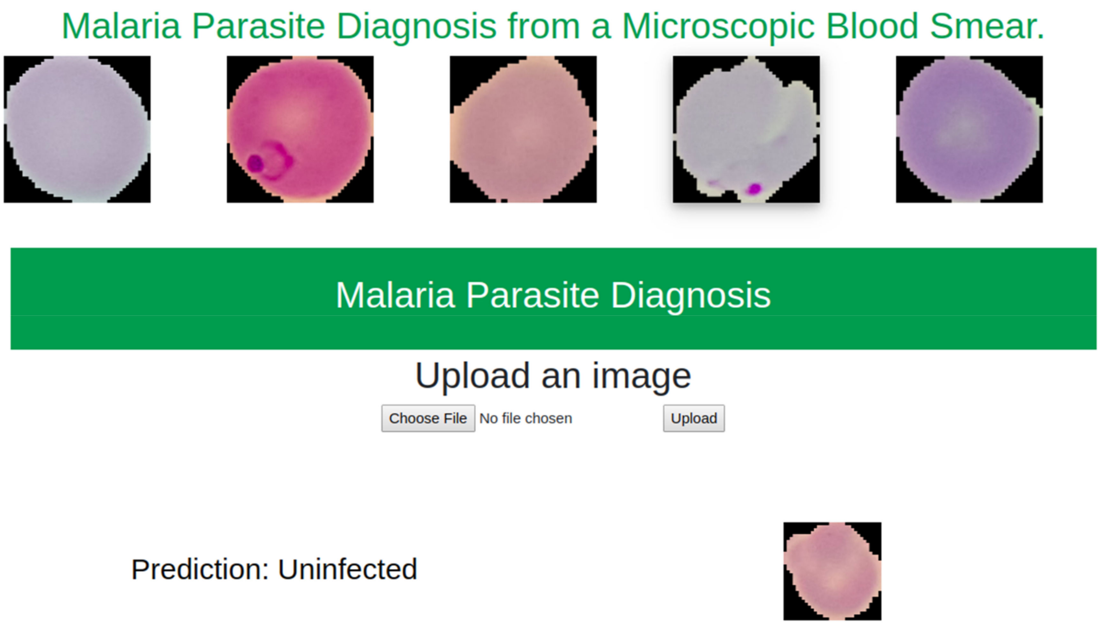

Malaria Parasite Diagnosis from a Microscopic Blood Smear.
Malaria Parasite Diagnosis
Upload an image
Choose File
No file chosen
Upload
Prediction: Uninfected

Figure 16. Model behavior on web based service.

6. Discussion

In this article, an efficient and highly accurate model is developed that detects malarial parasites from microscopic blood smear. In order to assess the suitability of deploying the model in power constrained devices, we conducted a series of experiments. Out of these experiments, we found that autoencoder outperforms other approaches. The reported accuracy of the autoencoder model is 0.9923 whereas, to the best of our knowledge, the best model performance, in terms of accuracy, was reported to be 0.9952 (Shown in Table 7). Rajaraman et al. used an ensemble of VGG-19 and SqueezeNet model to achieve this performance. However, their model lacks severely in terms of efficiency which makes them unsuitable to be deployed in power constrained devices. VGG-19 alone requires 19.6 billion flops whereas our model required a total of nearly 4600 flops. It clearly shows that our model is thousands times more efficient compared to the best reported performances.

Table 7. Performance comparison.

Method	Accuracy	Sensitivity	Specificity	F1 Score	Precision
Ross et al., (2006) [73]	0.730	0.850	-	-	-
Das et al., (2013) [74]	0.840	0.981	0.689	-	-
SS. Devi et al., (2016) [75]	0.9632	0.9287	0.9679	0.8531	-
Zhaohuiet al., (2016) [54]	0.9737	0.9699	0.9775	0.9736	0.9736
Bibin et al., (2017) [31]	0.963	0.976	0.959	-	-
Gopakumar et al., (2018) [58]	0.977	0.971	0.985	-	-
S. Rajaraman et al., (2019) [40]	0.995	0.971	0.985	-	-
Proposed Model	0.9923	0.9952	0.9917	0.9922	0.9892

The model developed can not only diagnose malarial parasites efficiently but also maintains high accuracy. Experimented model with image 28 × 28 achieves an accuracy of 0.9951 (Showed in Table 5) which is almost equal to the reported best performance. Considering the image quality, we proposed the model experimented on image 32 × 32 but this did not compromise too much accuracy (only 0.0029). From Table 5; Table 7, it is clearly evident that our model performs significantly well in terms of Sensitivity, Specificity and Precision compared with the current state of the art performances.

To show the efficiency and compatibility of our model, we deployed our model in both web-based and smartphone-based platforms. We also tested on different resource constrained smartphone-based systems as shown in Table 6. From the table it is evident that our model has compatibility to perform inference on any smartphone that is available in the market. To perform inference, it typically takes less than a second. In existing reported works, some of the model cannot even be deployed in any portable devices whereas our model can perform inference offline in nearly all market available power constrained devices. It clearly indicates the possibility of developing portable power constrained malaria parasite diagnostic systems for undeveloped, resources restricted, unprivileged communities.

7. Conclusions and Future Work

This paper presents multiple classification models for malaria parasite detection which take into consideration not only classification accuracy but also aim to be computationally efficient. In the process, we conducted a series of experiments including that of general training, distillation training and autoencoder training resulting in a comparison of 10 different models. From these the best performing model achieved an accuracy of 99.5% when trained using an Autoencoder based training method on 28 × 28 images, which is comparable to the performance reported by Rajaraman et al. We find a comparable accuracy of 99.23% when we trained on 32 × 32 images. We chose to do practical evaluations using this latter model due to the slightly higher image resolution with negligible difference accuracy. It is worth mentioning that this model requires only about 4600 flops compared to over 19.6 billion flops required for the model found in previously published work. Practical validation of model efficiency was performed by deploying the model in 10 different mobile phones of varying computational capacity and a server-backed web application. Data gathered from these environments show that the model can be used to perform inference under 1 s per sample in both offline (mobile only) and online (web application) mode. These inference speeds coupled with the high classification accuracy lead us to believe that this work can play a part towards building a fully automated system for malaria parasite detection which may be useful in resource-constrained areas in the foreseeable future.

Supplementary Materials: The following are available online at http://www.mdpi.com/2075-4418/10/5/329/s1; Figure S1: Predictions results where X-axis represents ground truth value and Y-axis represents predictions of the model: (**a**) depicts predicted result of 64 Augmentation and Distillation (**b**) depicts predicted result of 32 Distillation and Augmentation results, Figure S2: Effects of resizing images: (**a**) Depicts the quality of (32, 32) images (**b**) Depicts the quality of (28, 28) images, Figure S3: Some instances of falsely predicted data. X-axis is the original labeled data and Y-axis is model prediction.

Author Contributions: Conceptualization, K.M.F.F., J.F.T., N.M. and T.R.; Methodology, K.M.F.F.; Data curation, M.R.A.S. and J.F.T.; Formal Analysis and Investigation, K.M.F.F., N.M. and T.R.; Writing-original draft preparation, K.M.F.F., J.F.T. and M.R.A.S.; Writing- review and editing, J.F.T., T.R., N.M. and S.M.; Software, K.M.F.F. and M.R.A.S.; Validation, K.M.F.F., T.R. and N.M.; Supervision T.R., N.M., S.M. All authors have read and agreed to the publish version of the manuscript.

Funding: The research received no external funding.

Conflicts of Interest: There is no conflict of interest.

Appendix A. (Additional Result)

We have performed cross dataset validation as a method of evaluating the generalization of the trained model. For this purpose, we processed the images of another publicly available dataset [72] to obtain images of single blood cells from blood smear images. The method of data preprocessing, experimental details and result analysis have been explained in the following sections.

Appendix A.1. Dataset

In this dataset there are 655 Giemsa-Stained microscopic images of thin blood smear. The images were captured using a conventional light microscope. The stained objects are classified in two classes. One of the two classes of images have a presence of malarial parasites like *P. vivax*, *P. ovale*, *P. falciparum*, *P. malariae*. Of the 655 images, 109 were confirmed to contain some blood cells with malaria parasites. For our cross-dataset validation, we worked on these 109 images. Figure A1 depicts one of the images of 109 image.

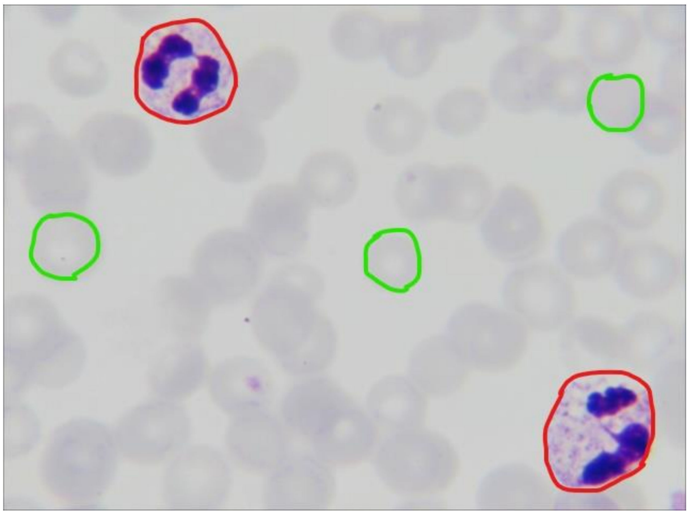


Figure A1. Microscopic Blood smear images of the dataset: Red color marked cells are the parasitized cells; Green color marked cells are some instances of uninfected cell.

Appendix A.2. Experimental Details

As the model is trained to make predictions on images of single cells, the blood smear images needed to be segmented. Segmentation was done manually as well as by using the Watershed [76] algorithm. For both cases, a selection of resulting segmented cell images was manually annotated to indicate presence/absence of malaria parasite.

In the case of manual segmentation to match the environment of the trained model, we changed the background of each image to black. As all the cells of a blood smear image does not contain malarial parasites, we manually selected the cells that contain malarial parasites and to achieve balance in the evaluation set we selected randomly an equal number of uninfected cells from the same blood smear. Using this process, we curated an evaluation set of 392 images with an equal distribution of infected and non-infected blood cells (can be found in [77]).

For automatic segmentation, we used the Watershed algorithm to perform segmentation. The Watershed algorithm is typically used to segment images when two regions of interest are close to each other [78]. The RGB images were transformed to gray scale and binarized using the OTSU method [79]. Later we performed morphological transformation (opening) [80] to reduce noise on the image and calculated the local peaks. Finally, we performed connected component analysis [81] on these local peaks and applied the Watershed algorithm. This process resulted in getting, on average, 96 segmented images per blood smear image. Due to the shortcomings of the segmentation process, some cells were segmented in multiple parts. When annotating the segmented images, we found 291 segmented images with a presence of a malaria parasite. Please note, some of these are not complete cells, due to the errors of the segmentation process. As before, to balance the data we selected 291 segmented images which do not contain malaria parasites (can be found in [82]).

Table A1. Segmentation Table.

Process	Manual	Automatic
Number of Blood smears	109	109
Background change	Black	-
Average segmentation on per cell	3	96
Finally Annotate cells	392	582
Number of Parasitized cells	196	291
Number of Uninfected cells	196	291

Appendix A.3. Result Analysis

Table A2 summarizes the performance of the model on the two sets of segmented images. The corresponding confusion matrices are shown in Figure A2. On both sets of segmented images, the performance is comparable (98.79% for automatic and 99.74% for manual), and in cases better than the performance obtained (99.23%) on the test partition of the dataset it was trained on. While the metrics are slightly lower for the automatically segmented evaluation set, as mentioned earlier this can mainly be attributed to the deficiencies of the very simple segmentation approach used. However, as cell segmentation is not in scope of this work, the performance on the manually segmented set is indicative of the model capacity when used with a mature segmentation process. These results, particularly the results obtained from the manually segmented evaluation set strongly argue for the generalizability of the trained model as it was not trained on this new collection, but still is able to perform at a similar level of accuracy and effectiveness.

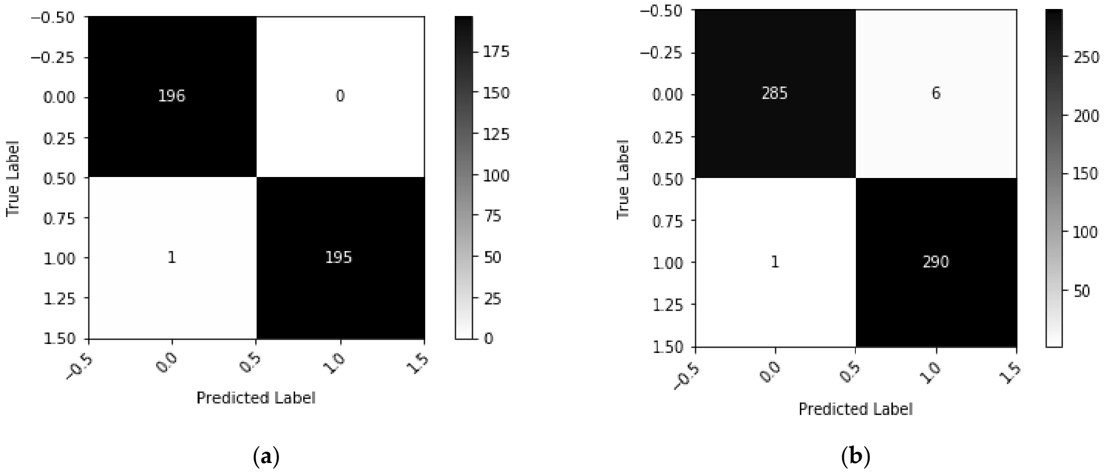


Figure A2. Confusion matrix of both experiments: (**a**) tested on manually segmented data; (**b**) tested on automatically segmented data.

Table A2. Result on Cross-Validation data.

Method	Test Accuracy	Test Loss	F1 Score	Precision	Sensitivity	Specificity
Manual	0.9974	0.0087	0.9974	1.00	0.9948	1.00
Automatic	0.9879	0.1183	0.9980	0.9797	0.9965	0.9793

References

1. Fact Sheet about Malaria. 2019. Available online: https://www.who.int/news-room/fact-sheets/detail/malaria (accessed on 29 December 2019).
2. Fact Sheet about MALARIA. 2020. Available online: https://www.who.int/news-room/fact-sheets/detail/malaria (accessed on 12 April 2020).
3. Schellenberg, J.A.; Smith, T.; Alonso, P.; Hayes, R. What is clinical malaria? Finding case definitions for field research in highly endemic areas. *Parasitol. Today* **1994**, *10*, 439–442. [CrossRef]
4. Tarimo, D.; Minjas, J.; Bygbjerg, I. Malaria diagnosis and treatment under the strategy of the integrated management of childhood illness (imci): Relevance of laboratory support from the rapid immunochromatographic tests of ict malaria pf/pv and optimal. *Ann. Trop. Med. Parasitol.* **2001**, *95*, 437–444. [CrossRef]
5. Díaz, G.; González, F.A.; Romero, E. A semi-automatic method for quantification and classification of erythrocytes infected with malaria parasites in microscopic images. *J. Biomed. Inform.* **2009**, *42*, 296–307. [CrossRef] [PubMed]
6. Ohrt, C.; Sutamihardja, M.A.; Tang, D.; Kain, K.C. Impact of microscopy error on estimates of protective efficacy in malaria-prevention trials. *J. Infect. Dis.* **2002**, *186*, 540–546. [CrossRef] [PubMed]
7. Anggraini, D.; Nugroho, A.S.; Pratama, C.; Rozi, I.E.; Iskandar, A.A.; Hartono, R.N. Automated status identification of microscopic images obtained from malaria thin blood smears. In Proceedings of the 2011 International Conference on Electrical Engineering and Informatics, Bandung, Indonesia, 17–19 July 2011; pp. 1–6.
8. Di Rubeto, C.; Dempster, A.; Khan, S.; Jarra, B. Segmentation of blood images using morphological operators. In Proceedings of the 15th International Conference on Pattern Recognition, ICPR-2000, Barcelona, Spain, 3–7 September 2000; Volume 3, pp. 397–400.
9. Di Ruberto, C.; Dempster, A.; Khan, S.; Jarra, B. Morphological image processing for evaluating malaria disease. In *International Workshop on Visual Form*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 739–748.
10. Mehrjou, A.; Abbasian, T.; Izadi, M. Automatic malaria diagnosis system. In Proceedings of the 2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), Tehran, Iran, 13–15 February 2013; pp. 205–211.
11. Alam, M.S.; Mohon, A.N.; Mustafa, S.; Khan, W.A.; Islam, N.; Karim, M.J.; Khanum, H.; Sullivan, D.J.; Haque, R. Real-time pcr assay and rapid diagnostic tests for the diagnosis of clinically suspected malaria patients in bangladesh. *Malar. J.* **2011**, *10*, 175. [CrossRef] [PubMed]
12. Masanja, I.M.; McMorrow, M.L.; Maganga, M.B.; Sumari, D.; Udhayakumar, V.; McElroy, P.D.; Kachur, S.P.; Lucchi, N.W. Quality assurance of malaria rapid diagnostic tests used for routine patient care in rural tanzania: Microscopy versus real-time polymerase chain reaction. *Malar. J.* **2015**, *14*, 85. [CrossRef] [PubMed]
13. Wongsrichanalai, C.; Barcus, M.J.; Muth, S.; Sutamihardja, A.; Wernsdorfer, W.H. A review of malaria diagnostic tools: Microscopy and rapid diagnostic test (rdt). *Am. J. Trop. Med. Hyg.* **2007**, *77* (Suppl. 6), 119–127. [CrossRef]
14. Pinheirob, V.E.; Thaithongc, S.; Browna, K.N. High sensitivity of detection of human malaria parasites by the use of nested polymerase chain reaction. *Mol. Biochem. Parasitol.* **1993**, *61*, 315–320.
15. Ramakers, C.; Ruijter, J.M.; Deprez, R.H.L.; Moorman, A.F. Assumption-free analysis of quantitative real-time polymerase chain reaction (pcr) data. *Neurosci. Lett.* **2003**, *339*, 62–66. [CrossRef]
16. Snounou, G.; Viriyakosol, S.; Jarra, W.; Thaithong, S.; Brown, K.N. Identification of the four human malaria parasite species in field samples by the polymerase chain reaction and detection of a high prevalence of mixed infections. *Mol. Biochem. Parasitol.* **1993**, *58*, 283–292. [CrossRef]
17. World Health Organization. *New Perspectives: Malaria Diagnosis*; Report of a jointwho/usaid informal consultation, 25–27 october 1999; World Health Organization: Geneva, Switzerland, 2000.

18. How Malaria RDTs Work. 2020. Available online: https://www.who.int/malaria/areas/diagnosis/rapid-diagnostic-tests/about-rdt/en/ (accessed on 9 April 2020).
19. WHO. New Perspectives: Malaria Diagnosis. Report of a Joint WHO/USAID Informal Consultation (Archived). 2020. Available online: https://www.who.int/malaria/publications/atoz/who_cds_rbm_2000_14/en/ (accessed on 9 April 2020).
20. Obeagu, E.I.; Chijioke, U.O.; Ekelozie, I.S. Malaria Rapid Diagnostic Test (RDTs). *Ann. Clin. Lab. Res.* **2018**, *6*. [CrossRef]
21. Available online: https://www.who.int/malaria/areas/diagnosis/rapid-diagnostic-tests/generic_PfPan_training_manual_web.pdf (accessed on 6 April 2020).
22. Microscopy. 2020. Available online: https://www.who.int/malaria/areas/diagnosis/microscopy/en/ (accessed on 9 April 2020).
23. Mathison, B.A.; Pritt, B.S. Update on malaria diagnostics and test utilization. *J. Clin. Microbiol.* **2017**, *55*, 2009–2017. [CrossRef] [PubMed]
24. Rajaraman, S.; Antani, S.K.; Poostchi, M.; Silamut, K.; Hossain, M.A.; Maude, R.J.; Jaeger, S.; Thoma, G.R. Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images. *PeerJ* **2018**, *6*, e4568. [CrossRef] [PubMed]
25. Quinn, J.A.; Nakasi, R.; Mugagga, P.K.; Byanyima, P.; Lubega, W.; Andama, A. Deep convolutional neural networks for microscopy-based point of care diagnostics. In Proceedings of the Machine Learning for Healthcare Conference, Los Angeles, CA, USA, 19–20 August 2016; pp. 271–281.
26. Poostchi, M.; Silamut, K.; Maude, R.J.; Jaeger, S.; Thoma, G. Image analysis and machine learning for detecting malaria. *Transl. Res.* **2018**, *194*, 36–55. [CrossRef]
27. Anggraini, D.; Nugroho, A.S.; Pratama, C.; Rozi, I.E.; Pragesjvara, V.; Gunawan, M. Automated status identification of microscopic images obtained from malaria thin blood smears using Bayes decision: A study case in Plasmodium falciparum. In Proceedings of the 2011 International Conference on Advanced Computer Science and Information Systems, Jakarta, Indonesia, 17–18 December 2011; pp. 347–352.
28. Yang, D.; Subramanian, G.; Duan, J.; Gao, S.; Bai, L.; Chandramohanadas, R.; Ai, Y. A portable image-based cytometer for rapid malaria detection and quantification. *PLoS ONE* **2017**, *12*, e0179161.
29. Arco, J.E.; Górriz, J.M.; Ramírez, J.; Álvarez, I.; Puntonet, C.G. Digital image analysis for automatic enumeration of malaria parasites using morphological operations. *Expert Syst. Appl.* **2015**, *42*, 3041–3047. [CrossRef]
30. Das, D.K.; Maiti, A.K.; Chakraborty, C. Automated system for characterization and classification of malaria-infected stages using light microscopic images of thin blood smears. *J. Microsc.* **2015**, *257*, 238–252. [CrossRef]
31. Bibin, D.; Nair, M.S.; Punitha, P. Malaria parasite detection from peripheral blood smear images using deep belief networks. *IEEE Access* **2017**, *5*, 9099–9108. [CrossRef]
32. Mohanty, I.; Pattanaik, P.; Swarnkar, T. Automatic detection of malaria parasites using unsupervised techniques. In Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering, Palladam, India, 16–17 May 2018; pp. 41–49.
33. Yunda, L.; Ramirez, A.A.; Millán, J. Automated image analysis method for p-vivax malaria parasite detection in thick film blood images. *Sist. Telemática* **2012**, *10*, 9–25. [CrossRef]
34. Ahirwar, N.; Pattnaik, S.; Acharya, B. Advanced image analysis based system for automatic detection and classification of malarial parasite in blood images. *Int. J. Inf. Technol. Knowl. Manag.* **2012**, *5*, 59–64.
35. Hung, J.; Carpenter, A. Applying faster r-cnn for object detection on malaria images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 56–61.
36. Tek, F.B.; Dempster, A.G.; Kale, I. Parasite detection and identification for automated thin blood film malaria diagnosis. *Comput. Vis. Image Underst.* **2010**, *114*, 21–32. [CrossRef]
37. Ranjit, M.; Das, A.; Das, B.; Das, B.; Dash, B.; Chhotray, G. Distribution of plasmodium falciparum genotypes in clinically mild and severe malaria cases in orissa, india. *Trans. R. Soc. Trop. Med. Hyg.* **2005**, *99*, 389–395. [CrossRef] [PubMed]
38. Sarmiento, W.J.; Romero, E.; Restrepo, A.; y Electrónica, D.I. Colour estimation in images from thick blood films for the automatic detection of malaria. In: Memoriasdel IX Simposio de Tratamiento de Señales, Im ágenes y Visión artificial. *Manizales* **2004**, *15*, 16.

39. Romero, E.; Sarmiento, W.; Lozano, A. Automatic detection of malaria parasites in thick blood films stained with haematoxylin-eosin. In Proceedings of the III Iberian Latin American and Caribbean Congress of Medical Physics, ALFIM2004, Rio de Janeiro, Brazil, 15 October 2004.
40. Rajaraman, S.; Jaeger, S.; Antani, S.K. Performance evaluation of deep neural ensembles toward malaria parasite detection in thin-blood smear images. *PeerJ* **2019**, *7*, e6977.
41. Corrected Malaria Data—Google Drive. 2019. Available online: https://drive.google.com/drive/folders/10TXXa6B_D4AKuBV085tX7UudH1hINBRJ?usp=sharing (accessed on 29 December 2019).
42. Linder, N.; Turkki, R.; Walliander, M.; Mårtensson, A.; Diwan, V.; Rahtu, E.; Pietikäinen, M.; Lundin, M.; Lundin, J. A malaria diagnostic tool based on computer vision screening and visualization of plasmodium falciparum candidate areas in digitized blood smears. *PLoS ONE* **2014**, *9*, e104855.
43. Opoku-Ansah, J.; Eghan, M.J.; Anderson, B.; Boampong, J.N. Wavelength markers for malaria (plasmodium falciparum) infected and uninfected red blood cells for ring and trophozoite stages. *Appl. Phys. Res.* **2014**, *6*, 47.
44. Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 37–52.
45. LeCun, Y.; Bengio, Y. Convolutional networks for images, speech, and time series. *Handb. Brain Theory Neural Netw.* **1995**, *3361*, 1995.
46. Dong, Y.; Jiang, Z.; Shen, H.; Pan, W.D.; Williams, L.A.; Reddy, V.V.; Benjamin, W.H.; Bryan, A.W. Evaluations of deep convolutional neural networks for automatic identification of malaria infected cells. In Proceedings of the 2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI), Orlando, FL, USA, 16–19 February 2017; pp. 101–104.
47. LeCun, Y. Lenet-5, Convolutional Neural Networks. URL. Available online: http://yann.lecun.com/exdb/lenet (accessed on 20 May 2015).
48. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, Nevada, 3–6 December 2012; pp. 1097–1105.
49. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
50. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv Prepr.* **2014**, arXiv:1409.1556.
51. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
52. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
53. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
54. Liang, Z.; Powell, A.; Ersoy, I.; Poostchi, M.; Silamut, K.; Palaniappan, K.; Guo, P.; Hossain, M.A.; Sameer, A.; Maude, R.J.; et al. Cnn-based image analysis for malaria diagnosis. In Proceedings of the 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Shenzhen, China, 15–18 December 2016; pp. 493–496.
55. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
56. Rosado, L.; Da Costa, J.M.C.; Elias, D.; Cardoso, J.S. Automated detection of malaria parasites on thick blood smears via mobile devices. *Procedia Comput. Sci.* **2016**, *90*, 138–144. [CrossRef]
57. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
58. Gopakumar, G.P.; Swetha, M.; Sai Siva, G.; SaiSubrahmanyam, G.R.K. Convolutional neural network-based malaria diagnosis from focus stack of blood smear images acquired using custom-built slide scanner. *J. Biophotonics* **2018**, *11*, e201700003. [CrossRef] [PubMed]
59. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. *Learning Internal Representations by Error Propagation*; Tech. rep.; California Univ San Diego La Jolla Inst for Cognitive Science: San Diego, CA, USA, 1985.

60. Sakurada, M.; Yairi, T. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis ACM, Gold Coast, Australia, 2 December 2014; p. 4.
61. Geng, J.; Fan, J.; Wang, H.; Ma, X.; Li, B.; Chen, F. High-resolution sar image classification via deep convolutional autoencoders. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2351–2355.
62. Sun, W.; Shao, S.; Zhao, R.; Yan, R.; Zhang, X.; Chen, X. A sparse auto-encoder-based deep neural network approach for induction motor faults classification. *Measurement* **2016**, *89*, 171–178.
63. Gemulla, R.; Nijkamp, E.; Haas, P.J.; Sismanis, Y. Large-scale matrix factorization with distributed stochastic gradient descent. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, 13–17 August 2011; pp. 69–77.
64. Denoeux, T. A k-nearest neighbor classification rule based on dempster-shafer theory. *IEEE Trans. Syst. Man Cybern.* **1995**, *25*, 804–813. [CrossRef]
65. Agarap, A.F. An architecture combining convolutional neural network (cnn) and support vector machine (svm) for image classification. *arXiv Prepr.* **2017**, arXiv:1712.03541.
66. Niu, X.X.; Suen, C.Y. A novel hybrid cnn–svm classifier for recognizing handwritten digits. *Pattern Recognit.* **2012**, *45*, 1318–1325.
67. Zheng, Y.; Jiang, Z.; Xie, F.; Zhang, H.; Ma, Y.; Shi, H.; Zhao, Y. Feature extraction from histopathological images based on nucleus-guided convolutional neural network for breast lesion classification. *Pattern Recognit.* **2017**, *71*, 14–25.
68. Buciluǎ, C.; Caruana, R.; Niculescu-Mizil, A. Model compression. In Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining ACM, Philadelphia, PA, USA, 20–23 August 2006; pp. 535–541.
69. Hinton, G.; Vinyals, O.; Dean, J. Distilling the knowledge in a neural network. *arXiv Prepr.* **2015**, arXiv:1503.02531.
70. Gracelynxs/Malaria-Detection-Model. 2019. Available online: https://github.com/gracelynxs/malaria-detection-model (accessed on 30 December 2019).
71. Bengio, Y. Rmsprop and equilibrated adaptive learning rates for nonconvex optimization. *arXiv Prepr.* **2015**, arXiv:1502.04390.
72. Tek, F.B.; Dempster, A.G.; Kale, I. Computer vision for microscopy diagnosis of malaria. *Malar. J.* **2009**, *8*, 153. [CrossRef] [PubMed]
73. Ross, N.E.; Pritchard, C.J.; Rubin, D.M.; Duse, A.G. Automated image processing method for the diagnosis and classification of malaria on thin blood smears. *Med. Biol. Eng. Comput.* **2006**, *44*, 427–436. [CrossRef] [PubMed]
74. Das, D.K.; Ghosh, M.; Pal, M.; Maiti, A.K.; Chakraborty, C. Machine learning approach for automated screening of malaria parasite using light microscopic images. *Micron* **2013**, *45*, 97–106. [CrossRef] [PubMed]
75. Devi, S.S.; Sheikh, S.A.; Talukdar, A.; Laskar, R.H. Malaria infected erythrocyte classification based on the histogram features using microscopic images of thin blood smear. *Ind. J. Sci. Technol.* **2016**, *9*. [CrossRef]
76. Najman, L.; Schmitt, M. Watershed of a continuous function. *Signal Process.* **1994**. [CrossRef]
77. Update Your Browser to Use Google Drive—Google Drive Help. 2020. Available online: https://drive.google.com/drive/folders/1etEDs71yLqoL9XpsHEDrSpUGwqtrBihZ?usp=sharing (accessed on 3 May 2020).
78. Preim, B.; Botha, C. *Visual Computing for Medicine*, 2nd ed.; Morgan Kaufmann: Burlington, MA, USA, 2013.
79. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. ManCybern.* **1979**, *9*, 62–66. [CrossRef]
80. Parvati, K.; Rao, P.; Mariya Das, M. Image segmentation using gray-scale morphology and marker-controlled watershed transformation. *Discret. Dyn. Nat. Soc.* **2008**, *2008*, 384346. [CrossRef]
81. Chai, B.B.; Vass, J.; Zhuang, X. Significance-linked connected component analysis for wavelet image coding. *IEEE Trans. Image Process.* **1999**, *8*, 774–784. [CrossRef]
82. Update Your Browser to Use Google Drive—Google Drive Help. 2020. Available online: https://drive.google.com/drive/folders/17YnJIMJIzOga5sC6jUEsexqZACs2n0Nw?usp=sharing (accessed on 3 May 2020).