

# Technical Report: Deep Learning-Based Malaria-Infected Cell Classification

## 1. Introduction

Malaria is a life-threatening disease caused by parasites transmitted through the bites of infected mosquitoes. Prompt and accurate diagnosis of malaria is crucial for effective treatment and disease management. In this report, we detail our approach to developing a deep learning-based model for classifying malaria-infected cells using microscopic images.

## 2. Dataset Description

The dataset used for training and evaluation consists of microscopic images of blood smear samples, containing both malaria-infected and uninfected cells. The dataset is divided into a training set and a test set, each containing images of various magnifications and stainings. Because the link you provided me that didn't work so that I had to download this dataset

## 3. Preprocessing

Before training the model, we preprocessed the images to ensure uniformity and compatibility with the chosen deep learning architecture. The preprocessing steps included resizing the images to a standard size of 224x224 pixels and normalization to scale pixel values between 0 and 1.

## 4. Model Selection

We opted for the VGG19 architecture, a widely used convolutional neural network (CNN) architecture known for its excellent performance on image classification tasks. The VGG19 model was pretrained on the ImageNet dataset, providing a solid foundation for feature extraction.

## 5. Transfer Learning

To leverage the knowledge learned by the VGG19 model on the ImageNet dataset, we employed transfer learning. We initialized the VGG19 model with pre-trained weights and fine-tuned it on our malaria-infected cell dataset. By freezing the convolutional layers and adding custom fully connected layers on top, we adapted the model to the specific task of malaria-infected cell classification.

## 6. Data Augmentation

To augment the training dataset and improve the model's generalization ability, we applied various data augmentation techniques using the Keras ImageDataGenerator. These techniques included random shearing, zooming, and horizontal flipping of images.

## 7. Model Training

The model was trained using the augmented training dataset and validated using the test dataset. We utilized the Adam optimizer with a categorical cross-entropy loss function to minimize the

classification error. The training process involved iterating over multiple epochs, with performance metrics monitored at each epoch.

## 8. Evaluation

After training, we evaluated the model's performance on the test dataset by calculating accuracy metrics and validation/testing accuracy comes out to be 94.78 percent . Additionally, we visualized the training and validation loss curves to assess the model's convergence and potential overfitting.

## 9. Findings

Our experiments demonstrated that the deep learning model achieved high accuracy in classifying malaria-infected and uninfected cells. The model's performance was robust across different magnifications and stainings of the microscopic images. The use of transfer learning and data augmentation proved effective in improving the model's generalization ability and mitigating overfitting.

## 10. Model Saving:

Once trained, the model is saved to a file named 'model\_vgg19.h5' for future use and deployment.

## 11. Conclusion

In conclusion, our approach to developing a deep learning-based malaria-infected cell classification model yielded promising results. By leveraging transfer learning, data augmentation, and the VGG19 architecture, we successfully built a robust model capable of accurately identifying malaria-infected cells. Further optimization and validation on larger datasets and real-world samples could enhance the model's utility in clinical settings.

Streamlit-Based Malaria-Infected Cell Classification App

### 1. Introduction

In this report, we detail the development of a Streamlit-based application for classifying malaria-infected cells using deep learning. The application allows users to upload microscopic images of blood smear samples and receive predictions on whether the cells are infected with malaria or not.

### 2. Data Preprocessing

The application accepts images in common formats such as JPG, PNG, and JPEG. Upon upload, the images are resized to a standard size of 224x224 pixels to match the input size expected by the pre-trained VGG19 model. Additionally, the images are preprocessed using the VGG19-specific preprocessing function to ensure compatibility with the model.

### 3. Model Loading

The pre-trained VGG19 model used for classification is loaded from a specified path. The model was previously trained on a large dataset, making it capable of capturing intricate features relevant to cell classification tasks.

#### 4. Streamlit App Design

The Streamlit application is designed with a simple user interface, featuring a file uploader widget for image input. Upon uploading an image, the application displays the uploaded image along with a caption. The preprocessed image is then fed into the loaded model for prediction.

#### 5. Prediction and Display

The model predicts the class label of the uploaded image, indicating whether the cell is infected with malaria or not. The predicted class label is displayed to the user, providing instant feedback on the classification result.

#### 6. Conclusion

The Streamlit-based malaria-infected cell classification application provides a user-friendly interface for users to quickly obtain predictions on uploaded microscopic images. By leveraging deep learning and pre-trained models, the application offers accurate and efficient classification capabilities, facilitating early diagnosis and treatment of malaria. Further enhancements could include incorporating additional features such as probability scores and visualization of model predictions for improved interpretability. Initially I face many problems while train the data set on vs code and save the model . that's why I trained and save my model through google colab and uploaded to vs code then I run streamlit app