Malaria Cell Infection Detection

Introduction:

Malaria is a life-threatening disease caused by the Plasmodium parasite, transmitted to humans through the bite of infected Anopheles mosquitoes. Early and accurate diagnosis is crucial for effective treatment and controlling the spread of the disease. This project focuses on the analysis of cell images to detect the presence of malaria parasites in red blood cells. The dataset used for this analysis is the "Malaria Cell Images Dataset," which consists of infected and uninfected cell images.

Dataset URL: https://www.kaggle.com/datasets/iarunava/cell-images-for-detecting-malaria/data

Dataset Details:

The "Malaria Cell Images Dataset" contains 27,558 images, categorized into two folders:

- Infected: Contains images of red blood cells infected with the malaria parasite. (13.8k Images)
- Uninfected: Contains images of healthy, uninfected red blood cells. (13.8k Images)

The dataset was obtained from the official NIH (National Institutes of Health) website. It has been made available for download to facilitate research and development in the field of malaria detection, as downloading from the NIH website can be slow.

Target Variable:

- Infected (or) Uninfected

Methodology:

1. **Data Preprocessing**

   The dataset is pre-processed to prepare it for training CNN model. Key preprocessing steps include:

   - To create a balanced dataset, we selected an equal number of samples from both "Infected" and "Uninfected" classes. This step is crucial to ensure that the models are not biased towards the majority class.
   - Image resizing to a uniform size of 128x128 pixels.
   - Conversion of images to grayscale.
   - Normalization of pixel values to a range between 0 and 1.

   The data is split into training, validation, and testing sets to train and evaluate machine learning models effectively.

2. **Data Augmentation**

   To improve the model's robustness and generalization, data augmentation techniques are applied using the ImageDataGenerator. These techniques include rotation, width and height shifting, shearing, zooming, horizontal and vertical flipping, and filling missing values.

3. **Convolutional Neural Networks (CNNs)**

   Two CNN-based models are developed for malaria cell detection: one without an autoencoder and one with an autoencoder.

   1. CNN without Autoencoder
      - Input layer with 128x128 pixel images.
      - Convolutional layers with max-pooling to capture features.
      - Flatten layer to convert feature maps into a vector.
      - Fully connected layers for classification.
      - Output layer with sigmoid activation for binary classification (infected or uninfected).

   2. CNN with Autoencoder
      - Encoder: Extracts features from input images.
      - Decoder: Reconstructs images for autoencoder training.
      - Classification part: Uses encoded features for classification.

   Both models are compiled with the Adam optimizer and binary cross-entropy loss function for binary classification. Training is performed for 20 epochs.

Results & Discussion:

The performance of both models was compared based on the evaluation metrics. Here are the key findings:

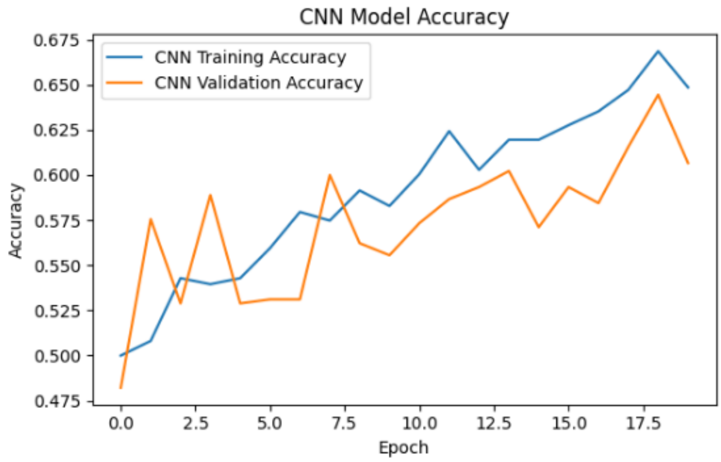
1. CNN without Autoencoder achieved 62% of Training Accuracy, 60% of Validation Accuracy, and **58% of Test Accuracy**.
2. CNN with Autoencoder achieved 68% of Training Accuracy, 69% of Validation Accuracy, and **66% of Test Accuracy**. The accuracy/loss plots for this model can be found in the visualization section.

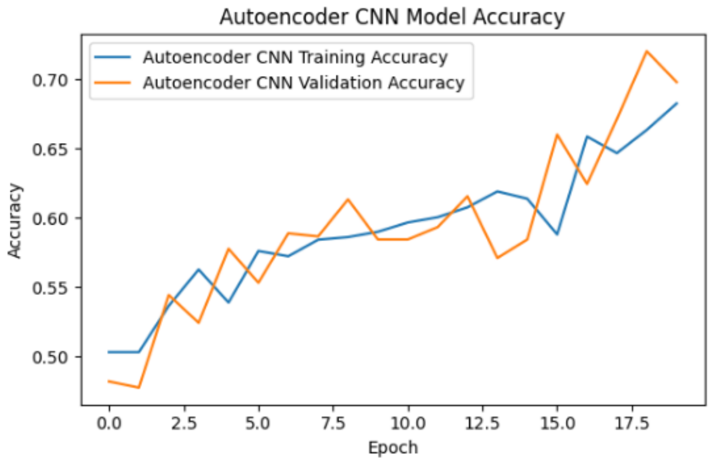
Model	Accuracy
CNN without Autoencoder	58%
CNN with Autoencoder	66%

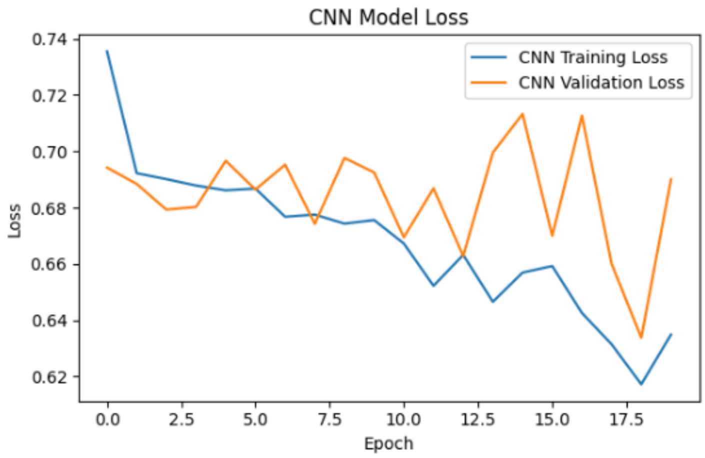
The results indicate that the CNN with an autoencoder in the early layers outperformed the CNN without an autoencoder in terms of test accuracy. This suggests that the incorporation of an autoencoder for feature extraction can be beneficial in image classification tasks. The improved performance of the CNN with an autoencoder may be attributed to the ability of the autoencoder to capture and represent relevant features from the input images. These learned features could be more discriminative for classification tasks, especially in scenarios where data is noisy or has complex patterns.

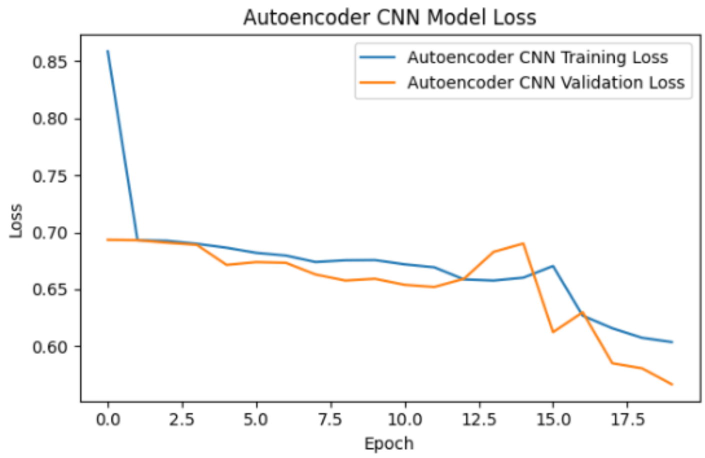
However, it is essential to consider that the performance improvement may not be significant in all scenarios. The choice to use an autoencoder should be based on the specific characteristics of the dataset and problem at hand.

Visualization:









Conclusion:

This project investigated the impact of integrating autoencoders into CNN models for image classification. The results suggest that incorporating an autoencoder in the early layers of a CNN can lead to improved classification performance, as demonstrated by the higher test accuracy of the CNN with an autoencoder.

The CNN without an autoencoder achieved a test accuracy of 58%, while the CNN with an autoencoder achieved a significantly higher test accuracy of 66%. This significant improvement in accuracy suggests that integrating an autoencoder in the early layers of the CNN enhances the model's ability to classify malaria-infected and uninfected red blood cells accurately.

It is important to note that the effectiveness of an autoencoder may vary depending on the dataset and problem domain. Further research and experimentation are recommended to explore the applicability of autoencoders in different image classification tasks.

Overall, this study contributes to our understanding of the role of autoencoders in enhancing the performance of CNNs for image classification, highlighting their potential in improving model accuracy and robustness.