Slide1:

Heading: Automating Malaria Pathology using Computer Vision:

Script: Malaria diagnosis traditionally relies on microscopic inspection of stained blood smears. Trained pathologists identify the presence of Plasmodium parasites within red blood cells, which appear as distinct ring-like or irregular structures. Differentiating infected from uninfected cells is crucial, but the process is time-consuming, labor-intensive, and prone to human error.

Slide2:

Heading: The Malaria Endemic:

Script: This choropleth map highlights the global burden of malaria. In 2000, 108 countries reported active malaria endemic zones, shown in red, green, and yellow. By 2023, the number reduced to 83 countries, still marked in red. The most severely impacted regions remain Africa, South Asia, and parts of Latin America.

Slide3:

Heading: Training Dataset:

Script: This slide shows examples from our training dataset. We used 27,556 labeled cell images from the TensorFlow “malaria” dataset, each tagged as uninfected or parasitized. Parasitized cells reveal dark, irregular Plasmodium structures inside, while uninfected cells appear clear and uniform. These distinguishing features are what our CNN learns to detect.

Slide4:

Heading: CNN LeNet Model Architecture:

Script: As the task is relatively simple, we implemented CNN LeNet architecture for malaria detection. Input images were first resized to 224×224×3 pixels before entering the network. The model begins with two convolutional layers (6 and 16 filters), each followed by batch normalization and max pooling to extract features. Kernel size of 5 was used as there are not many complicated features to learn in the Convolution layers.

The output of the Convolution layers are next flattened and passed through dense layers of 100 and 10 neurons with ReLU activation. Finally, a sigmoid output layer classifies cells as parasitized or uninfected, enabling efficient binary classification.

Slide5:

Heading: Model Summary, Parameter Counts and Optimal Hyper Parameters:

Script: The LeNet model contains about 4.5 million trainable parameters. The first convolutional layers are lightweight, with 456 and 2,416 parameters respectively. Most parameters come from the fully connected dense layer with 100 units, contributing over 4.49 million. Batch normalization layers add a few hundred each, while the final dense layers with 10 and 1 neurons add just over 1,000 combined. **This distribution shows that parameter complexity is dominated by the dense layers rather than convolutional filters**.

We achieved the best performance with an 80-10-10 train-test-validation split, batch size of 32, Adam optimizer at a 0.01 learning rate, and binary cross-entropy loss. Early stopping limited training to 8–9 epochs, preventing overfitting.

The final model is saved in Anodiam’s Github repository and the link is shared below.