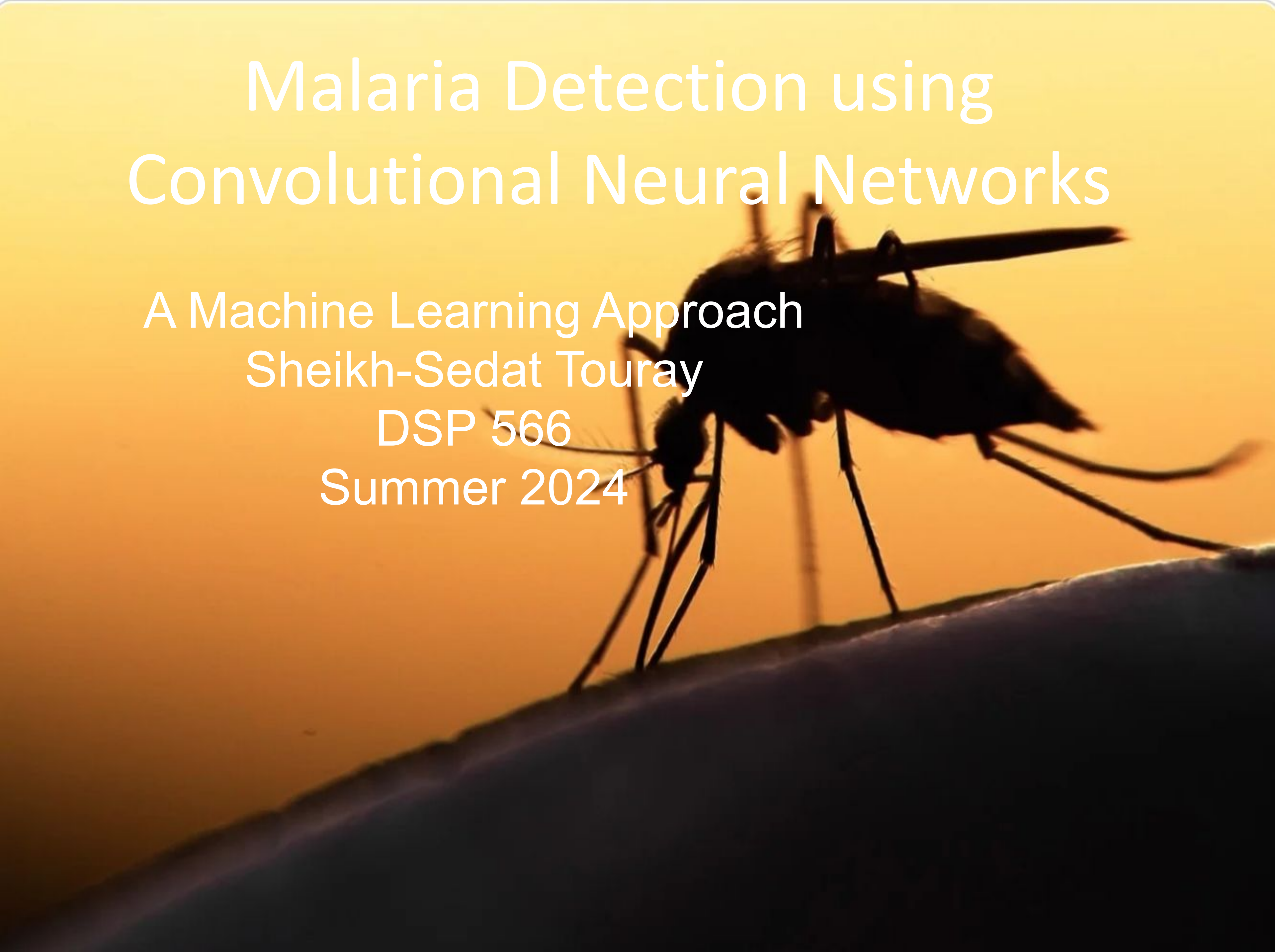
Malaria Detection using Convolutional Neural Networks

A Machine Learning Approach

Sheikh-Sedat Touray

DSP 566

Summer 2024



Introduction

- **Overview of Malaria**: Malaria is a highly infectious disease caused by the bite from female Anopheles Mosquito. The parasites multiply in the liver and infect the red-blood cells leading to flu like symptoms, and may cause organ failure and even death.
- One of the importance of early detection is that it prevents severe complications like cerebral malaria, organ failure or even death.
- **Objective**:
  The goal of building Convolutional Neural Network (CNN) models to classify cell images as either parasitized or uninfected is to automate the process of detecting malaria infection in cell images. The primary goal is for a timely and accurate diagnosis of blood samples that uninfectious and those that are parasitized for early treatment. Also, manual examination of blood smears under a microscope by trained medical professionals is time-consuming and requires significant expertise. A CNN model can automate this process, allowing for faster and more consistent analysis, especially in areas with limited medical resources.

Dataset Overview

- Dataset Description:
- Source: TensorFlow Malaria Dataset.
  27,558 training samples.
  Two categories: Parasitized and Uninfected
  images.

CNN Model Architecture

- Layer Details:
- Conv2D, MaxPooling, Flatten, Dense, Dropout layers.
- Output: Sigmoid activation for binary classification.
- Model Summary:
  Total parameters: 3,304,769

CNN Model Training

- Training Details:
- Batch size: 32
- Epochs: 10
- Optimizer: Adam
- Loss Function: Binary Cross-Entropy
- Epoch Performance:
  Graph of accuracy and loss over epochs.
  Key metrics: Training accuracy ~98.12%, Validation accuracy ~95.14%.

CNN Model Results

- Final Performance:
- Validation Accuracy: 95.16%
- Validation Loss: 0.2127
- Observations:
  Discuss overfitting, stability of accuracy, and areas of improvement.

VGG16 Transfer Learning Architecture

- VGG16 Architecture:
- Pre-trained on ImageNet.
- Custom layers added: Flatten, Dense, Dropout.
- Model Summary:

  Total parameters: 14,977,089

  Trainable parameters: 262,401

VGG16 Model Training

- Training Details:
- Batch size: 32
- Epochs: 5
- Optimizer: Adam
- Loss Function: Binary Cross-Entropy
- Epoch Performance:
  Graph of accuracy and loss over epochs.
  Key metrics: Training accuracy ~90.11%, Validation accuracy ~93.31%.

VGG16 Model Results

- Final Performance:
- Validation Accuracy: 93.48%
- Validation Loss: 0.1878
- Observations:
  Transfer learning with VGG16 effectively leverages pre-trained features, achieving comparable accuracy to the CNN from scratch with less training data. However, the CNN from scratch can be more efficient and tailored to specific datasets, offering faster convergence with fewer resources..

Comparison of Models

- Performance Comparison:
- CNN vs. VGG16:

Accuracy: CNN (from Scratch): **~95.16%** validation accuracy; VGG16 (Transfer Learning): **~93.48%** validation accuracy

Loss: CNN: Final loss of **0.2127**, indicating good fit; VGG16: Lower validation loss of **0.1878**

Generalization Ability: CNN: Strong generalization with minimal overfitting; VGG16: Slightly higher tendency to overfit but strong generalization

Computational Resources: **CNN: ~3.3M parameters,** less demanding ;**VGG16: >14.7M** parameters, more resource-intensive

Training Time: **CNN: Faster training (~10 epochs in ~1,000 mins)**; VGG16: Slower training **(~5 epochs in ~2,000 mins)**

Model Complexity:CNN: Simpler, tailored to the specific dataset; VGG16: Deep, complex, captures intricate features

Conclusion

- Summary:
- My CNN model was very effective with a high accuracy and stability, VGG16's performance was good but not better than my CNN, importance of data augmentation allows you to rotate the data and focus on fillmode.
- Future Work:
  Explore other models and architectures e.g Xception-like, experiment with more data, and optimize hyperparameters.

Questions

- Thank you! Any questions?

References

- Citations: References, datasets, or frameworks used in the project.

- https://www.tensorflow.org/datasets/catalog/malaria
- https://www.news-medical.net/health/Mosquito-borne-Diseases.aspx
- https://www.worksheetsplanet.com/what-is-malaria/
- https://builtin.com/machine-learning/vgg16