Project Report: Advanced Image Classification Model for Chess Pieces

Objective

My primary objective was to develop a sophisticated image classification model capable of identifying different chess pieces using the Chessman Image Dataset provided. My submission aims to cover the entire machine learning pipeline, including data preprocessing, model training, evaluation, deployment, and containerization. I employed advanced techniques such as data augmentation and regularization to enhance model performance. The final deliverables included a REST API and a Streamlit interface for model deployment, both containerized using Docker.

Project Developement Lifecycle

1. Data Preparation

I used the Chessman Image Dataset for this project. The dataset underwent several preprocessing techniques to improve the model's generalizability:

Normalization: I rescaled the pixel values were rescaled to the range [0, 1].
Resizing: All images were resized to a uniform size of 150x150 pixels.
Data Augmentation: I employed techniques such as rotation, flipping, and scaling to increase the diversity of the training data using Imagen.
Splitting: The dataset was split into training, validation, set 80-20 ratio.

2. Model Selection and Enhancement

For the initial model, I chose Convolutional Neural Networks (CNNs) due to their effectiveness in image recognition tasks. Later, I employed transfer learning with a pre-trained VGG16 model to leverage pre-existing knowledge from the ImageNet dataset. This approach helped achieve better performance with limited data.
Base Model: VGG16 pre-trained on ImageNet. I futher enhanced the model by unfreezing the top layers and incorporating techniques such as dropout (0.5) and batch normalization to improve accuracy and reduce overfitting.

3. Training

I trained the model on the Chessman dataset with the following configurations:
Optimizer: Adam with a learning rate of 0.0001.The model was trained on the Chessman dataset using early stopping and model checkpointing to monitor training dynamics and save the best model. I finetuned the Hyperparameters using grid search to find the optimal settings. The model was trained for 30 epochs with a batch size of 16

4. Evaluation

I evaluated the model's performance using metrics such as accuracy, precision, recall, and F1-score, I also generated a confusion matrix analyze the model's performance across different chess piece categories.

Accuracy: Both training and validation accuracy showed significant improvement, stabilizing at around almost 90%.Training and validation loss decreased consistently, indicating proper model convergence. I Used a confuson matrics to identify misclassifications and areas for improvement.

Figure 1: Accuracy and Loss during Training

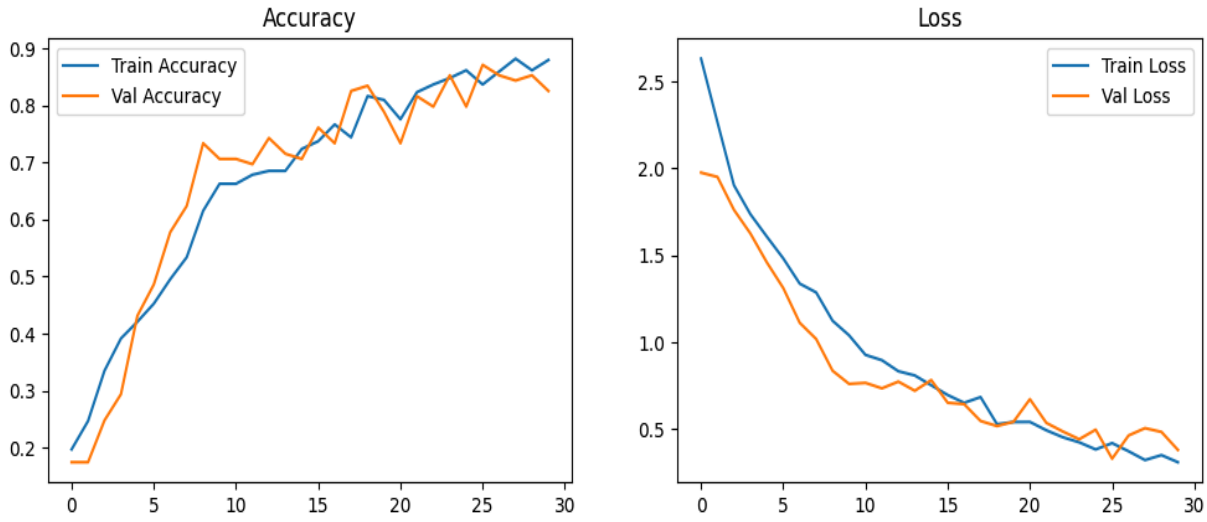
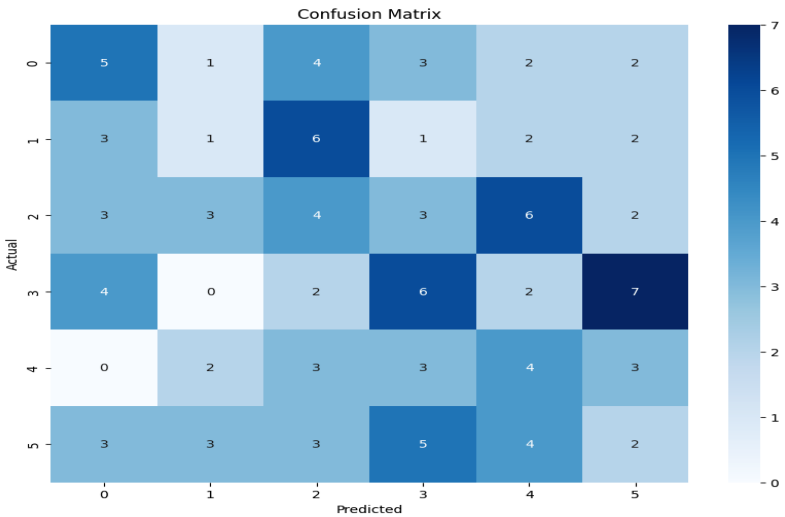


Figure 2: Confusion Matrix



5. REST API

I developed a REST API using FastAPI to serve the model and handle image classification requests.

Endpoints: /predict endpoint <accessed at /docs> to accept image uploads and return classification results.I implemented ronust error handling mechanisms to manage invalid inputs and server errors.

6. Streamlit Interface

I designed a user-friendly Streamlit interface to allow users to upload images and view classification results. Integration with API: The interface communicated with the FastAPI backend to fetch predictions.

7. Dockerization

I containerized the entire pipeline, including the model, API, and Streamlit interface, using Docker and Docker Compose. I also created Separate Dockerfiles for the API and Streamlit interface, i used a docker-compose.yml file was used to manage multi-container deployment. To ensure communication between the docker Containers I used Docker bridge network to facilitate that

8. GitHub and Documentation

I created a comprehensive GitHub repository with all source code, Docker files, and documentation.

README: Included setup instructions, usage guidelines, and troubleshooting tips.

Source Code: Organized into production grade directories for data preprocessing, model training, evaluation, and deployment and also tesing notebook.