Chess Recognition

The **Chess 2D Board Recognition** algorithm focused on developing an intelligent computer vision framework capable of detecting and classifying chess pieces on 2D digital boards using advanced AI techniques. To achieve this, a custom convolutional neural network (CNN) was trained specifically to interpret flat, stylized chessboard images. The system was designed to integrate with a broader visual understanding pipeline, allowing it to automatically analyze the layout of a chessboard and generate structured semantic information from it.

To enable accurate recognition, the dataset was created by cropping high-resolution chessboard images into 64 square segments corresponding to the standard 8×8 grid. Each cropped image was manually labeled into one of thirteen possible categories: *empty*, *white_pawn*, *white_rook*, *white_knight*, *white_bishop*, *white_queen*, *white_king*, *black_pawn*, *black_rook*, *black_knight*, *black_bishop*, *black_queen*, and *black_king*. The dataset was divided into training and validation subsets in an 80/20 ratio to ensure model generalization and prevent overfitting.

The CNN was implemented using the **ResNet-18** architecture, chosen for its strong balance between performance and computational efficiency. The network's final classification layer was replaced with a fully connected layer corresponding to the thirteen chess piece categories. Training was conducted for twenty epochs on 64×64 pixel images of the cropped squares. During this process, the model learned to distinguish between empty and occupied squares while accurately classifying each piece type. Upon completion, the trained network achieved high accuracy and was saved as **chess_cnn.pth** for later integration into the main recognition pipeline.

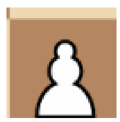
Once integrated, the CNN processes an input chessboard by dividing it into 64 regions, classifying each square, and mapping detected pieces to their respective coordinates (e.g., *white_pawn on a2*, *black_queen on d8*). The system expresses these results as structured semantic triplets, such as
{ "subject": "white_pawn", "predicate": "on", "object": "a2" },
which can then be used in downstream reasoning or knowledge graph modules to describe board states in a machine-interpretable format.

Overall, the CNN demonstrated reliable performance in identifying all major chess pieces across a variety of digital board designs. The system achieved strong consistency in both piece recognition and positional mapping, forming a robust foundation for higher-level reasoning tasks such as game-state reconstruction and semantic analysis of chessboard images.

Methodology with examples:

1. Images of chessboards from different websites were saved and cropped into 64 regions

Examples of resulting crops can be seen below:

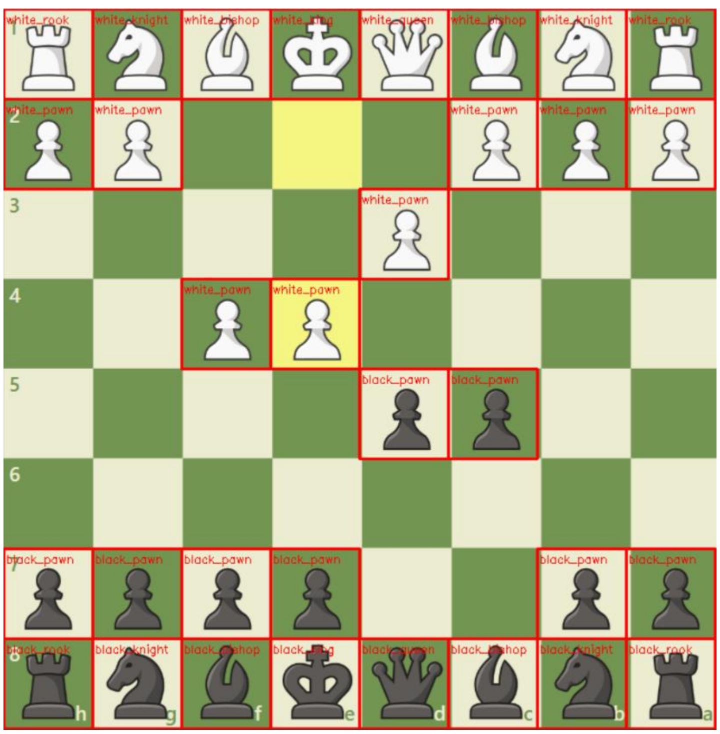
  

2. A simple CNN model was trained using 475 training images and 132 validation images.

```
✅ Found 475 training images and 132 validation images.
Classes: ['black_bishop', 'black_king', 'black_knight', 'black_pawn', 'black_queen', 'black_rook', 'empty', 'white_bishop', 'white_king', 'white_knight', 'w
hite_pawn', 'white_queen', 'white_rook']
Using device: cpu
Epoch 1/20 | Train Acc: 68.63% | Val Acc: 89.39% | Loss: 0.3280
Epoch 2/20 | Train Acc: 86.95% | Val Acc: 95.45% | Loss: 0.1258
Epoch 3/20 | Train Acc: 90.11% | Val Acc: 93.94% | Loss: 0.5277
Epoch 4/20 | Train Acc: 91.16% | Val Acc: 97.73% | Loss: 0.0876
Epoch 5/20 | Train Acc: 92.42% | Val Acc: 98.48% | Loss: 0.0657
Epoch 6/20 | Train Acc: 93.89% | Val Acc: 89.39% | Loss: 0.2033
Epoch 7/20 | Train Acc: 95.79% | Val Acc: 98.48% | Loss: 0.0825
Epoch 8/20 | Train Acc: 97.05% | Val Acc: 99.24% | Loss: 0.0328
Epoch 9/20 | Train Acc: 98.74% | Val Acc: 95.45% | Loss: 0.2073
Epoch 10/20 | Train Acc: 96.84% | Val Acc: 97.73% | Loss: 0.0804
Epoch 11/20 | Train Acc: 97.05% | Val Acc: 100.00% | Loss: 0.0093
Epoch 12/20 | Train Acc: 95.37% | Val Acc: 93.94% | Loss: 0.3129
Epoch 13/20 | Train Acc: 94.53% | Val Acc: 95.45% | Loss: 0.2553
Epoch 14/20 | Train Acc: 93.68% | Val Acc: 100.00% | Loss: 0.0091
Epoch 15/20 | Train Acc: 97.05% | Val Acc: 100.00% | Loss: 0.0032
Epoch 16/20 | Train Acc: 95.37% | Val Acc: 100.00% | Loss: 0.0021
Epoch 17/20 | Train Acc: 98.74% | Val Acc: 100.00% | Loss: 0.0013
Epoch 18/20 | Train Acc: 99.37% | Val Acc: 100.00% | Loss: 0.0004
Epoch 19/20 | Train Acc: 99.58% | Val Acc: 100.00% | Loss: 0.0015
Epoch 20/20 | Train Acc: 99.16% | Val Acc: 100.00% | Loss: 0.0019
```

3. The model was then tested on various chessboard images from different websites.


1
2
3
4
5
6
7
8
h
g
f
e
d
c
b
a


white_rook
white_knight
white_bishop
white_king
white_queen
white_bishop
white_knight
white_rook
white_pawn
white_pawn
white_pawn
white_pawn
white_pawn
white_pawn
white_pawn
white_pawn
black_pawn
black_pawn
black_pawn
black_pawn
black_pawn
black_pawn
black_pawn
black_pawn
black_rook
black_knight
black_bishop
black_king
black_queen
black_bishop
black_knight
black_rook
3
4
5
6
h
g
f
e
d
c
b
a