Journal Report 14

1/13/20-1/17/20

Kevin Fu
Computer Systems Research Lab
Period 1, White

Daily Log

Monday January 13

Tried adding class weights to address imbalanced dataset, found they slow down training significantly. (20 s to over 120 s per epoch.) Commented code.

Tuesday January 14

Gave up on class weights because of slow training speeds. Took 36 images of chessboards with randomly arranged non-pawn pieces, to manually balance dataset. Labelled a third of images.

Wednesday January 15

Discussed project during lunch with Dr. Gabor and Dr. White. Labelled another 12 images. Goal going forward: have fluid live demo before chess states (March).

Thursday January 16

While labelling last third of dataset, realized canny edge thresholding removes a lot of black pieces from being considered when on dark squares. Wrote script to remove some of empty class for more manual rebalancing of dataset. Started training on this dataset, reached 95% val acc in 60 epochs, but training cut short for time.

Timeline

Date	Goal	Met
Winter Goal (Dec 19)	Have a script that converts a chessboard image to a digital array using a piece-recognition CNN (75% accuracy)	Done
Jan 6	Reorganize and comment code, figure out how to load Keras models quicker, consider training empty/not-empty network	Code commented, waiting on hardware sticks, considered and decided against
Jan 13	Continue experimenting for better piece-recognition model	In progress
Jan 20	Finish experimenting, work on reducing real-world runtimes by cutting down squares model looks at (chess logic)	Not started
Jan 27	Separate input mechanism from rest of Winter Goal script, integrate new model/square-checking into Winter Goal script	Not started

Reflection

I'm glad I pivoted from using class weights to perfectly balance my dataset with math to just adding more images to reduce imbalance. The early returns have been promising, and I'm excited to fully train the model and see if it's less overzealous about guessing empty squares. I also modified my expanded dataset by cutting the 'empty' class in half, to further reduce the imbalance in my dataset; will try training a model on this as well. Next week, after training, I'll analyze my model's performance both on a case-by-case basis, as I've been doing, but also with a confusion matrix, as Dr. White suggested, to get an overall picture of my model's performance.

I'll also work on reducing the number of squares that the piece detection model needs to consider. First, I'll write a function that takes a position in a chess match and gives the full subset of possible next positions (without knowing whose turn it is), then collates that information to give which pieces could possibly be on each square. Then I can filter my CNN's results accordingly. I'll also look into color change detection, the Stockfish-based probability distribution detailed in the paper Kevin Chung is using, and consider the whole board. For example, it's impossible for there to be two kings of the same color on the board, or two white bishops on like colored-squares.