# Computer vision system to read a chess position

Samuel Rapier
Student Number: 680029868

November 1, 2021

Signed: __________________________________

I certify that all material in this dissertation which is not my own work has been identified.

**Abstract**

# 1 Introduction

Chess engines are used by players to analyse chess positions, indicating how to proceed and evaluating how strong a given position is. Chess engines can online be used on a computer and so require a position to be entered onto the computer.

The ability to recognise an over the board (OTB) chess position and convert it to an online chess position can be very helpful to players for both the ability to use a chess engine and to record chess games. Chess puzzles are a common way for players to practice and some players prefer to practice these on a real board. However, this limits the players ability to analyse positions using a chess engine without having to spend time copying out the position.

Recording chess games as they progress is a task most commonly done by writing each chess move by hand until the game concludes. In high level games, they may make use of electronic chess boards, also known as DTG boards, which detects the pieces and automatically records the moves played. These can be very expensive and thus not realistic for amateur players. Recording the game by photo or video and automatically converting the position to a digital version would be a much cheaper solution in these cases.

Identifying chess pieces and chess positions remains a complex computer vision problem despite the advances in Convolutional neural networks.

In this paper we will analyse previous methods of identifying chess positions and compare their approaches whilst looking into the methods with which this problem could be solved.

# Literature Review

# 2 Board detection

Identifying the location of a chess board is the first step towards understanding the chess position. This is a complex machine learning problem which can be approached using either edge detection or corner detection.

A chess board is composed of 64 squares in an 8x8 grid formation with alternating coloured squares. Thus it consists of 9 horizontal lines and 9 vertical lines which can be utilised by line-based approaches.

We must keep in mind that identifying the chessboard with pieces on the board can add extra complication since the chess pieces obscure vision of the board itself and potentially may obscure features of the board which are needed to confidently identify it.

## 2.1 Corner-based board detection

Corner detection approaches use first detect the corners of the chess board and often perform Hough transforms to identify the lines of the chess board, then locating to coordinates of the corners of each square on the chess board. This requires the chess board to be on a plain background to ensure that the lines detected are only those belonging to the chess board. Additionally, this requires the image to either be taken from a top-down view [2] or the board be empty of chess pieces to ensure that none of the corners are obstructed from view [1]. This approach works well for a game tracking solution, however, makes it difficult to take a snapshot of a position and determine the position, without first initialising the board.

Typical corner detection methods include SUSAN corner detector, Harris corner detector and template matching-based corner detector. According to Tam et al. [6] corner based approaches have a high tolerance against camera distortion since corners are less effected by warping. However, they are limited when corners are obscured and can be tricked by jagged lines caused by digitisation artifacts.

## 2.2 Line-based board detection

Common approaches to line-based begin with edge detection. Edge detection is performed by detecting significant changes in pixel brightness in grey-scale images as explained by D. Ziou et al. [3], some examples of these include Canny edge detection and Marr-Hildreth detectors. Hough transforms can then be used to identify the lines of the chessboard. Given that a chess board can be identified by 18 total lines and the orientation of half the lines will be perpendicular to the others makes them a popular approach [4, 5]. They are also deemed a more robust approach since the lines are less likely to be completely obscured and it is more resistant to noise since a line will have significantly higher number of pixels than an individual corner.

Tam et al. [6] used a line based approach, using Hough transforms to detect lines, they proceeded to use the geometry of a single square to extrapolate the locations of points of other squares.

A. De la Escalera [5] combined both a line-based approach and a corner-based approach to achieve good results at detecting the chessboard. However, as C. Danner et al. [4] identified, many irrelevant corners where being detected and when chess pieces are on the board, many corners could not be detected, further reducing the usefulness of the corner-based approach.

# 3 Piece Recognition

There are a number of game tracking applications which assume the starting positions of the chess pieces and detect the piece movements to build up the whole game. However, as previously stated, this would not work in our approach since we would like to be able to determine an unknown position from an individual image, without requiring the whole game to be played. Additionally, this approach would likely not work in other variations of chess, such as Fischer-Random [7] chess since the starting point of each of the pieces is randomised.

Therefore, we need an approach which will be able to classify each of the pieces on the board. Some techniques start by detecting the colour of the piece, and then use shape-descriptors to identify them fully []. A few more recent approaches have used machine learning classifiers to determine identify the pieces [].

## 3.1 Shape-descriptor identifiers

## 3.2 Classifier approaches

J. Ding [8] trained a classifier with gradient based feature descriptors. They experimented with both scale-invariant feature transform (SIFT) [] and histogram of oriented gradients (HOG) [] and determined that HOG performed better than SIFT since HOG is well-suited to detection problems. Their implementation produced piece detection accuracy of 95% and piece classification accuracy of 85%.

### 3.2.1 Experimental Few-shot learning approach

# Project Specification

# 4 Conclusion

## References


[1] J. Hack and P. Ramakrishnan. (2014). "CVChess: Computer Vision Chess Analytics"

[2] C. Koray and E. Sumer, "A Computer Vision System for Chess Game Tracking," presented at the 21st Computer Vision Winter Workshop, Rimske Toplice, Slovenia, 2016

[3] D. Ziou and S. Tabbone, "Edge Detection Techniques - An Overview" 1998

[4] C. Danner and M. Kafafy. (2015). "Visual Chess Recognition"

[5] A. De la Escalera and J. M. Armingol, "Automatic chessboard detection for intrinsic and extrinsic camera parameter calibration," Sensors, vol. 10, no. 3, pp. 20272044, 2010

[6] K. Y. Tam, J. A. Lay, and D. Levy, "Automatic grid segmentation of populated chessboard taken at a lower angle view," in Digital Image Computing: Techniques and Applications (DICTA), 2008. IEEE, 2008, pp. 294–299.

[7] https://en.wikipedia.org/wiki/Fischer_random_chess

[8] Ding, J. ChessVision: Chess board and piece recognition.