Ben-Gurion University of the Negev

Faculty of Engineering Science

School of Electrical and Computer Engineering

Dept. of Electrical and Computer Engineering

Digital Image Processing Project

Final Report

ChessVision - Chess Board and Pieces Recognition

Students	**Ariel Moshe, 312549660**
(ID & Name):	**Roy Schneider, 31253815**
	Ram Shirazi, 204272058
:Lecturer	**Dr. Tammy Riklin Raviv**
Submitting date:	**05/03/2021**

Table of Contents

List of Figures..2
Abstract 3
System Design..3
    Project setup 3
    Features 4
Algorithm Flow..4
    Calibration 4
    Piece Classification 5
    Move Tracking 6
    Main assumptions.. 7
    Challenges and Limitations... 7
Performance Analysis of Piece Classification 8
Conclusion.. 10
References... 11

List of Figures

Figure 1 - Selected board.. 3
Figure 2 - Full setup 3
Figure 3 - Chess pieces 4
Figure 4 - Board calibration of different orientations 5
Figure 5 - Hand detection 6
Figure 6 - Piece detection 6
Figure 7 – Game tracking.. 7
Figure 8 - Regular lightning 9
Figure 9 – Direct light... 9
Figure 10 – Poor lightning..10

Abstract

The recording of moves during a chess game is often a tedious manual task that impedes the flow of the game, especially in formats such as speed chess where manually recording moves hinders efficient use of time. In this project we create MATLAB program which record a chess game, identifies pieces and track moves in the game. The procedure divides to three stages: calibration, piece classification and move tracking. The users can use the program to identify the piece, place them on the board and initialize a game position. After the position is set the players can start playing, while the program will track the moves and display graphical board on live.

System Design

Project setup

The project setup includes iPhone 11 camera with 1980x1080 resolution that record the chess board during a game, from top view angle, connected to a laptop via Wi-Fi.
The camera is being held by desk mount phone arm and so the setup is mobile.
Generic three-dimensional pieces as used in competitions and gold-brown squares chess board with some decorations. Furthermore, a computer screen will show the graphical interface.

Figure 1 - Selected board

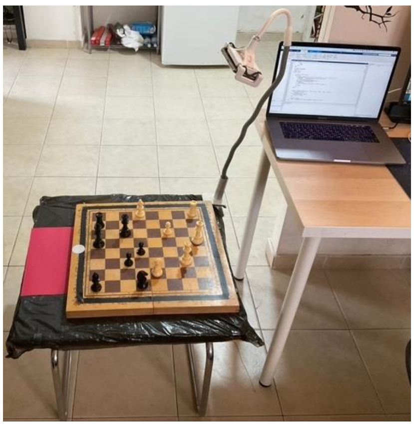

Figure 2 - Full setup

Figure 3 - Chess pieces

Features

- Pythonic GUI that updates during the game.
- Classification of all pieces on a designated location, marked with a simple white sticker on top of the board. Detecting the position on which the classified piece is placed.
- Establishing any initial game position (as long as it legal).
- Check for legality of initial game position after piece classification.
- Detect player's move and check for illegal moves.
- Navigate through algorithm stages using simple hand gesture.

Algorithm Flow

The algorithm consists of three main individual phases: calibration, piece classification and move tracking. Each phase is briefly explained during this section. The algorithm implemented using MATLAB program.

Calibration

The goal is to find the projective transform in order to have a top view of the board for simplifying the location calculations. We used built-in MATLAB function *detectCheckerboardPoints* which detect the 7x7 inner points of the live chess board capture using Hough lines intersections and Harris corner detection (red circles on figure 4). In addition to the points, a simple grid of reference points with same size is generated to find the transformation. The function struggles finding the right orientation with symmetric checkboards, since it depends on which dimension (rows or columns) is the major in the counting. To ensure that the right orientation is achieved, a white piece needed to be placed on A1 corner. Using the transformation that calculated, the corners of the board can be cropped squares, and using the mean of gray intensity of the squares the right orientation is decided.

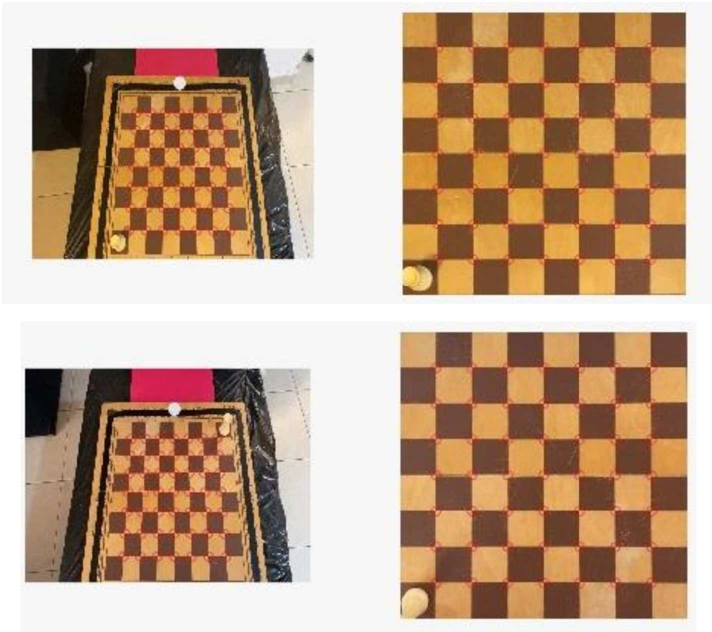

Figure 4 - Board calibration of different orientations

Piece Classification

During the calibration a white round sticker placed on the board is detected using Hough circles detection. This sticker marks a designated spot where the pieces are placed in when they are being classified. The player needs to place the pieces turn by turn on this spot during the classification state.

To classify the pieces, at first the video captures are being cropped in the region of the mark that calculated using the location and the radius of the detected circle together with the transformation, to extract image of the classified piece. The correlation between the cropped image with dataset of reference images computed, and the maximum of the correlation was extracted. In case the maximum correlation was above 85%, the piece was classified by the label of the reference image that corresponds to the maximum. To increase the performance robustness to lighting conditions the histogram of the cropped images was matched to each reference image, before the correlation calculation took place.

After the piece was classified, the player needs to place the piece on the board. The location is detected by finding the maximum mean of the absolute value of the difference between gray intensities of two images. One is taken before the piece was placed and the other after, that taken when the system identifies appearance of a hand and its disappearance right after, more details in next stage. The detection of the hand performed by thresholding the Lab color-space.

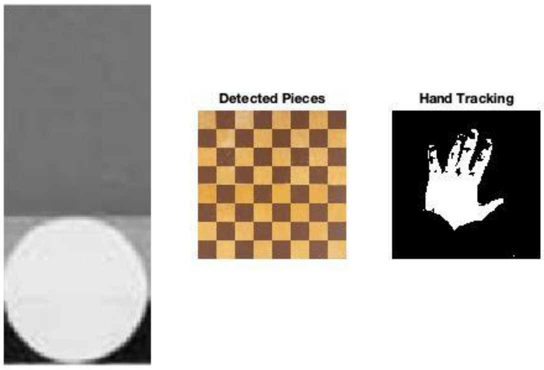

Figure 5 - Hand detection

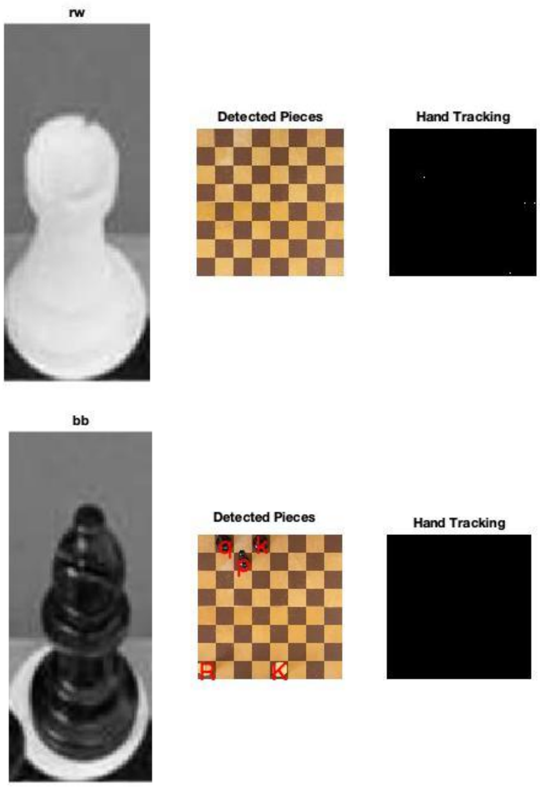

Figure 6 - Piece detection

Move Tracking

After the classification was done, one player should keep his hand on top of the board for 10 seconds to notify the system that everything in place and procced to move tracking stage. The players are asked which side is starting and the game begins. The system track changes using the means of the gray intensities of the difference image as described before. In more details, the straighten image of the board was extracted to a fixed size, so the squares of the board could be mapped precisely to constructing 8x8 matrix of the means. The system differentiates the images using hand tracking as in the piece classification stage and find the two maxima of the mean matrix. A move was detected between the locations corresponds to those maxima, if they both was above empiric threshold. The system validates each move and if a wrong move was played it notifies and won't continue the game tracking until the player will undo his move.

Figure 7 – Game tracking

Main assumptions

- The board environment is with reasonable lighting.
- The players should not wear any sleeves that may cover their arm because it disrupts the hand recognition.
- The recording angle and board location is fixed due to limitation of piece classification.
- Piece classification is done piece by piece at a designated spot.

Challenges and Limitations

The main challenge of this project was to detect the pieces. Detecting pieces right from a given board with any piece arrangement was unachieved task, so the detection was done by placing piece in a specific location and comparing the capture to a reference image. This method is very restricted and has raised challenges and problems:

- The pieces have to be placed precisely on the designated spot. In addition, the recording angle must fit the angle in which the data set images were capture.
- Asymmetric piece, such as king and knight should be in a fixed orientation also.
- Due to unprecise positioning, the system might classify the wrong piece.
- Successful detection in different lightning condition was a big challenge, but we manage to modify the detection to be robust to lighting condition.

In addition, in move tracking stage the system encounters problem in location detection of tall piece that intersects two squares and informs wrong locations. This limits the black player to place tall pieces on the last row only. This drawback can be taken care of by setting the camera angle to 90-degree (top view).

Performance Analysis of Piece Classification

Our first approach was to train SVM classifier based on SURF features and also as a second attempt using HOG features with different parameters as mentioned in [2], but since the images of the pieces were too small and the pieces are with smooth textures, together with different shooting perspective along the board and pieces that hiding each other the extracted features were not reliable enough to fit a model. As a result, we turn to a different approach using one reference image in one designated spot. This model can be considered as over-fitted and analyze number of detections is irrelevant, since any piece image that different from the references won't be detected. Thus, we thought about creative ways to analyze the piece classification performance.

We test system robustness to lightning condition. We try to classify the pieces in different environments: regular lightning, poor lightning, and direct light that causes glare. In 2 extreme cases, the system detected all pieces without missing anyone. The histogram matching appeared useful for system robustness to light condition. Next are shown examples of the captured pieces images in different lighting conditions where the left one is the captured pieces, the middle one is the reference image, and the right is the histogram matched image.

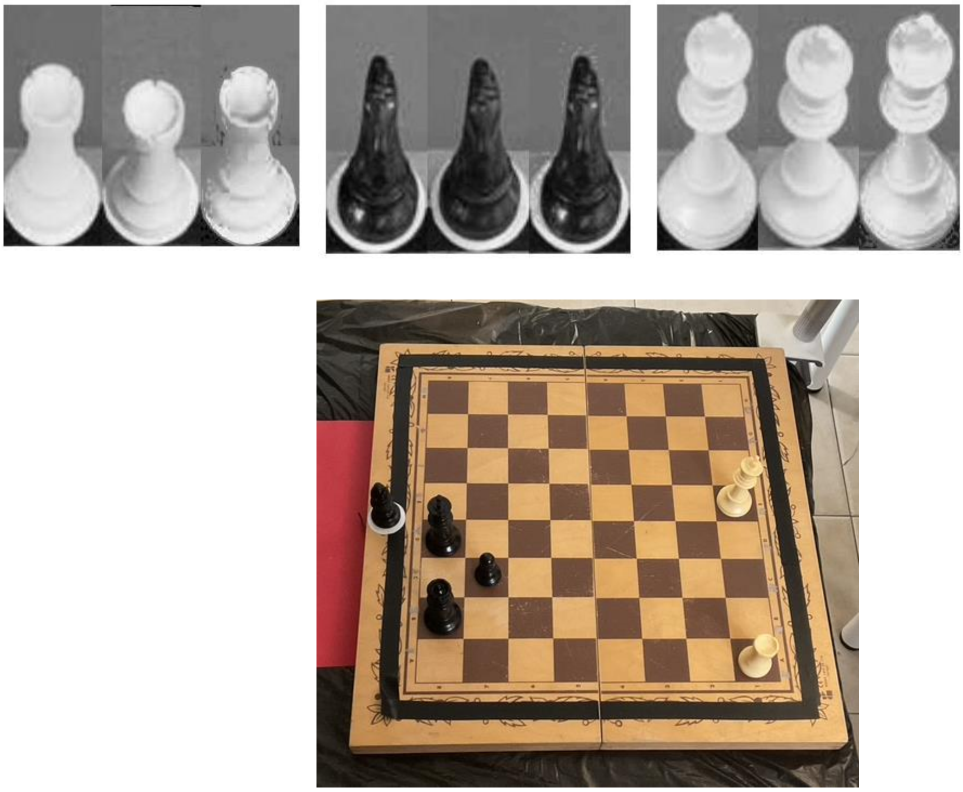

Figure 8 - Regular lightning

Figure 9 – Direct light

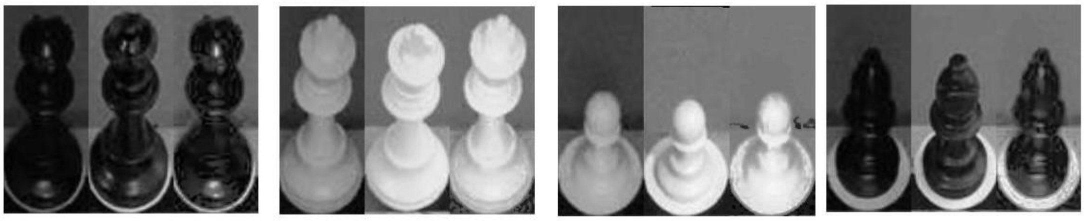

Figure 10 – Poor lightning

Conclusion

In this project, we present an approach for 3D chess piece recognition using correlation between images and histogram matching.

It seems that the algorithm deals well with the extreme environments such as poor and directed light. In addition, it also deals with captures taken from slightly different angles and orientation of some pieces such as bishop and rook. Still, its performance depends on very strict conditions, and if the fixed camera angle or the orientation of asymmetric piece such as king and night will be changed the detection will fail.

Using the machine learning approach to differentiate between three-dimensional pieces, resulted in low accuracy that is not practical for chess game application. Furthermore, to improve performance and robustness, neural networks can be used with more comprehensive data set to deal with occlusions, different angles camera and orientation as mentioned in [4].

References

[1] A. de la Escalera and J. M. Armingol, "Automatic Chessboard Detection for Intrinsic and Extrinsic Camera Parameter Calibration," vol. 10, no. 3, pp. 2027-2044, 2010, doi: 10.3390/s100302027.

[2] J. Ding, "ChessVision: Chess Board and Piece Recognition.".

[3] "Chess Recognition Using Computer Vision." May 30, 2017.

[4] M. A. Czyzewski, A. Laskowski, and S. Wasik, "Chessboard and chess piece recognition with the support of neural networks," Aug. 2017, [Online]. Available: https://arxiv.org/abs/1708.03898.