Chapter 1

Introduction

1.1 Context survey

Chess recognition is a problem in computer vision whereby an algorithm is tasked with recovering the configuration of pieces from an image of a chessboard. Early work on chess recognition in the 1990s focused on extracting typeset games from printed material [1]. In recent years, the problem of parsing two-dimensional chess images has effectively been solved using conventional machine learning techniques [2] and even deep learning [3], [4]. However, recognising chess positions from physical chessboards as opposed to artificial two-dimensional images poses a much more interesting and challenging problem that finds practial application in chess-playing robots, augmented reality, and aiding amateur chess players[1].

Chess robots Initial research into chess recognition emerged from the development of chess robots that included a camera to detect the human opponent's moves from a top-down overhead perspective. The difficulty of distinguishing between chess pieces from a bird's-eye-view due to their similarity is noted in many papers; as a result, chess robots typically implement a three-way classification system that for every square attempts to determine whether it contains a piece, and if so, the piece's colour. Various approaches have been explored including employing manual thresholding [6]–[9] and clustering [10] in different colour spaces, as well as differential imaging (classifying based on the per-pixel difference between two images) [11], [12]. Although the *Gambit* robot proposed by Matuszek *et al.* [13] does not require a bird's-eye view over the chessboard and uses a depth camera to more reliably detect the occupancy of each square, it still employs the three-way classification strategy using a linear support vector machine (SVM) to determine the piece colour.

Chess move recording Several techniques for recording chess moves from video footage have been proposed that follow a similar three-way occupancy and colour classification scheme, both from a top-down perspective [5], [14] as well as from a camera positioned at an acute angle to the board [15]. However, in any

[1] Electronic chess sets are impractical and very costly [5], thus solutions for chess recognition using just a photo of an unmodified chess board are more compelling for amateur chess players.

three-way classification approach, the robot or move recorder requires knowledge of the previous board state in addition to its predictions for each square's occupancy and piece colour to deduce the last move. While this information is readily available to a chess robot or move recording software, this is not the case for a chess recognition system that should deduce the position from a single still image. Furthermore, these approaches experience severe shortcomings in terms of their inability to recover once a single move was predicted incorrectly and fail to identify promoted pieces[2] [6].

Single-image chess recognition A number of techniques have been developed to address the issue of chess recognition from a single image. Unlike move recording software or chess robots, it does not suffice to only determine the occupancy and colour of each square, but each piece must be identified. These techniques must implement a classification algorithm for each piece type (pawn, knight, bishop, queen, and king) of each colour which poses a significantly more difficult problem, attracting research mainly in the last five years. From a bird's-eye view, the pieces are nearly indestinguishable, so the photo is usually taken at an angle to the board. Ding [16] proposes a piece classifier that uses one-versus-rest SVMs trained on scale-invariant feature transform (SIFT) and histogram of oriented gradients (HOG) feature descriptors, achieving an accuracy of 85%. Danner and Kafafy [17] as well as Xie *et al.* [18] claim that SIFT and HOG provide inadequate features for the problem of piece classification due to the similarity in texture between chess pieces, and instead focus on the pieces' outlines. As such, Danner and Kafafy [17] use Fourier descriptors calculated for the pieces' contours, but this requires a manually-created database of piece silhouettes. Furthermore, they modify the board colours to red and green instead of black and white, in order distinguish the pieces from the board more easily[3]. On the other hand, Xie *et al.* [18] perform contour-based template matching with an interesting caveat: the camera angle is calculated based on the perspective transformation of the chessboard, and then depending on the angle, different templates are utilised for matching the chess pieces. As part of the same work, Xie *et al.* developed another approach that instead utilised convolutional neural networks (CNNs), but found that their original template-matching technique achieved superior results in terms of speed and accuracy in low-resolution images. However, it is important to note that their CNNs were trained on only 40 images per class and deep learning methods tend to excel when trained on larger datasets.

Chessboard detection A prerequisite to any chess recognition system is the ability to detect the location of the chessboard and each of the 64 squares. Once the four corner points have been established, finding the squares is trivial for pictures captured in bird's-eye view, and only a matter of a simple perspective transformation in the case of other camera positions. While finding the corner points of a chessboard is frequently used for automatic camera calibration due

[2]Piece promotion occurs when a pawn reaches the last rank, in which case the player must choose to promote to a queen, rook, bishop or knight. Evidently, a vision system that can only detect the piece's colour is unable to detect what it was promoted to.

[3]Similar board modifications have also been proposed as part of chess robots [8] and chess move trackers [5], but any such modification imposes an unreasonable constraint on normal chess games.

to the regular nature of the chessboard pattern [19], [20], techniques designed for this purpose tend to perform poorly when there are pieces on the chessboard that occlude lines or corners. Some of the aforementioned chess robots [10], [11], [14] as well as the single-image recognition system proposed by Danner and Kafafy [17] circumvent this problem entirely by prompting the user to interactively select the four corner points, but ideally a chess recognition system should be able to parse the position on the board without human intervention. Most approaches for automatic chess grid detection utilise either the Harris corner detector [8], [15] or a form of line detector based on the Hough transform [9], [12], [17], [21]–[24], although other techniques such as template matching [13] and flood fill [5] have been explored. In general, corner-based algorithms are unable to accurately detect grid corners when they are occluded by pieces, thus line-based detection algorithms appear to be the favoured solution. Such algorithms often take advantage of the geometric nature of the chessboard which allows to compute a perspective transformation of the grid lines that best matches the detected lines [15], [18], [21]. However, lines found in the background of the photo can often cause failure modes. A recent chess grid detection algorithm that is highly successful even on populated boards is descibed by Xie *et al.* in [24]. They apply several clustering algorithms on the lines detected via a Hough transform in order to find the horizontal and vertical grid lines belonging to the chessboard, and use this algorithm as a preprocessing step in their template-matching piece classification technique [18] described above.

Chess recognition using CNNs Since Xie *et al.* pioneered the use of CNNs in the domain of chess recognition from monocular images in 2018[4], a few more techniques have been developed that employ CNNs at various stages in the recognition pipeline. Czyzewski *et al.* [26] achieve an accuracy of 95% on chessboard detection from non-vertical camera angles by designing an iterative algorithm that gereates heatmaps over the input image representing the likelihood of each pixel being part of the chessboard. They then employ a CNN to refine the corner points that were found using the heatmap, outperforming the results obtained by Gonçalves *et al.* [10]. Furthermore, they compare a CNN-based piece classification algorithm to the SVM-based solution proposed by Ding [16] and find no notable improvement, but manage to obtain major improvements by implementing a probabilistic reasoning system that uses the open source Stockfish chess engine [27] as well as chess statistics [28]. Although reasoning techniques were already employed for refining the predictions of chess recognition systems before [17], [22], Czyzewski *et al.* demonstrate the potential of combining information obtained from a chess engine with large-scale chess statistics. Very recently, Mehta and Mehta [29] implemented an augmented reality app using the popular *AlexNet* CNN architecture introduced by Krizhevsky *et al.* [30], achieving promising results. Despite using an overhead camera perspective and not performing any techniques to ensure probable and legal chess positions, Mehta and Mehta achieve an end-to-end accuracy of 93% for the entire chessboard detection and piece classification pipeline.

[4]Wei *et al.* [25] developed a chess recognition system using a volumetric CNN one year previously, but this approach requires three-dimensional chessboard data obtained from a depth camera. Their approach achieved a per-class accuracy over 90% except for the “king” class, was trained on computer-aided design (CAD) models, and evaluated on real three-dimensional images (point clouds) of a chessboard.

Datasets The lack of adequate datasets for chess recognition has been recognised by many [16], [26], [29]. Although Czyzewski *et al.* [26] published a dataset of chessboard lattice points that are difficult to predict [31], large datasets – especially at the scale required for deep learning – are not available as of now. Using synthesised data in the training set is an efficient means of creating sizable datasets while minimising the manual annotation efforts [25], [26], [32]. Czyzewski *et al.* distort some input images in order to simulate different camera perspectives on the chessboard corners. However, a more promising method seems to be the use of three-dimensional models. Wei *et al.* [25] synthesise point cloud data for their volumetric CNN directly from three-dimensional chess models and Hou [32] use renderings of three-dimensional models as input. Yet Wei *et al.* [25]'s approach works only if the chessboard was captured with a depth camera and Hou [32] presents a chessboard recognition system using a simple artificial neural network (ANN) that is not convolutional and hence achieves an accuracy of only 72%.

Acronyms

ANN artificial neural network. 4

CAD computer-aided design. 3

CNN convolutional neural network. 2–4

HOG histogram of oriented gradients. 2

SIFT scale-invariant feature transform. 2

SVM support vector machine. 1, 2

Bibliography

[1] H. Baird and K. Thompson, 'Reading chess,' *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, pp. 552–559, Jun. 1990, ISSN: 1939-3539. DOI: `10.1109/34.56191`.

[2] I. M. Khater, A. S. Ghorab and I. A. Aljarrah, 'Chessboard recognition system using signature, principal component analysis and color information,' in *2012 Second International Conference on Digital Information Processing and Communications (ICDIPC)*, Jul. 2012, pp. 141–145. DOI: `10.1109/ICDIPC.2012.6257285`.

[3] A. Sameer, *Tensorflow_chessbot*, 6th Sep. 2020. [Online]. Available: `https://github.com/Elucidation/tensorflow_chessbot` (visited on 11/09/2020).

[4] A. Roy, *Chessputzer*, 12th Jul. 2020. [Online]. Available: `https://github.com/metterklume/chessputzer` (visited on 11/09/2020).

[5] V. Wang and R. Green, 'Chess move tracking using overhead RGB webcam,' in *2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013)*, Nov. 2013, pp. 299–304. DOI: `10.1109/IVCNZ.2013.6727033`.

[6] T. Cour, R. Lauranson and M. Vachette, 'Autonomous Chess-playing Robot,' École Polytechnique, Palaiseau, France, Jul. 2002. [Online]. Available: `http://www.timotheecour.com/papers/ChessAutonomousRobot.pdf`.

[7] D. Urting and Y. Berbers, 'MarineBlue: A low-cost chess robot,' in *IASTED International Conference Robotics and Applications, RA 2003, June 25-27, 2003, Salzburg, Austria*, M. H. Hamza, Ed., IASTED/ACTA Press, 2003, pp. 76–81.

[8] N. Banerjee, D. Saha, A. Singh and G. Sanyal, 'A Simple Autonomous Chess Playing Robot for playing Chess against any opponent in Real Time,' in *Proceedings of International Conference on Computational Vision And Robotics*, vol. 58, Bhubaneshwar, India: Interscience Research Network, Aug. 2012, pp. 17–22. [Online]. Available: `http://nandanbanerjee.com/files/ICCVR-08AUG12-011%20paper.pdf`.

[9] A. T.-Y. Chen and K. I.-K. Wang, 'Computer vision based chess playing capabilities for the Baxter humanoid robot,' in *2016 2nd International Conference on Control, Automation and Robotics (ICCAR)*, Apr. 2016, pp. 11–14. DOI: `10.1109/ICCAR.2016.7486689`.

[10] J. Gonçalves, J. Lima and P. Leitão, 'Chess robot system : A multi-disciplinary experience in automation,' in *9th Spanish Portuguese Congress on Electrical Engineering*, 2005. [Online]. Available: `https://bibliotecadigital.ipb.pt/handle/10198/1898` (visited on 17/09/2020).

[11] R. A. M. Khan and R. Kesavan, 'Design and development of autonomous chess playing robot,' *2014 IJISET - International Journal of Innovative Science, Engineering & Technology*, vol. 1, no. 1, 2014, ISSN: 2348-7968. [Online]. Available: `http://www.ijiset.com/v1s1/IJISET_V1_I1_01.pdf`.

[12] A. T.-Y. Chen and K. I.-K. Wang, 'Robust Computer Vision Chess Analysis and Interaction with a Humanoid Robot,' *Computers*, vol. 8, no. 1, p. 14, 1 Mar. 2019. DOI: `10.3390/computers8010014`.

[13] C. Matuszek, B. Mayton, R. Aimi, M. P. Deisenroth, L. Bo, R. Chu, M. Kung, L. LeGrand, J. R. Smith and D. Fox, 'Gambit: An autonomous chess-playing robotic system,' in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 4291–4297. DOI: `10.1109/ICRA.2011.5980528`.

[14] E. Sokic and M. Ahic-Djokic, 'Simple Computer Vision System for Chess Playing Robot Manipulator as a Project-based Learning Example,' in *2008 IEEE International Symposium on Signal Processing and Information Technology*, Dec. 2008, pp. 75–79. DOI: `10.1109/ISSPIT.2008.4775676`.

[15] J. Hack and P. Ramakrishnan, 'CVChess: Computer Vision Chess Analytics,' Stanford University, Win. 2014. [Online]. Available: `https://web.stanford.edu/class/cs231a/prev_projects_2015/chess.pdf`.

[16] J. Ding. (2016). 'ChessVision : Chess Board and Piece Recognition,' [Online]. Available: `/paper/ChessVision-%3A-Chess-Board-and-Piece-Recognition-Ding/08a4ae9ce38da309780d925e6b18e7906a12a3d9` (visited on 20/09/2020).

[17] C. Danner and M. Kafafy. (2015). 'Visual Chess Recognition,' [Online]. Available: `https://web.stanford.edu/class/ee368/Project_Spring_1415/Reports/Danner_Kafafy.pdf` (visited on 12/09/2020).

[18] Y. Xie, G. Tang and W. Hoff, 'Chess Piece Recognition Using Oriented Chamfer Matching with a Comparison to CNN,' in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2018, pp. 2001–2009. DOI: `10.1109/WACV.2018.00221`.

[19] A. De la Escalera and J. M. Armingol, 'Automatic Chessboard Detection for Intrinsic and Extrinsic Camera Parameter Calibration,' *Sensors*, vol. 10, no. 3, pp. 2027–2044, 3 Mar. 2010. DOI: `10.3390/s100302027`.

[20] S. Bennett and J. Lasenby, 'ChESS – Quick and robust detection of chess-board features,' *Computer Vision and Image Understanding*, vol. 118, pp. 197–210, 1st Jan. 2014, ISSN: 1077-3142. DOI: `10.1016/j.cviu.2013.10.008`.

[21] K. Tam, J. Lay and D. Levy, 'Automatic Grid Segmentation of Populated Chessboard Taken at a Lower Angle View,' in *2008 Digital Image Computing: Techniques and Applications*, Dec. 2008, pp. 294–299. DOI: `10.1109/DICTA.2008.40`.

[22] J. E. Neufeld and T. S. Hall, 'Probabilistic location of a populated chessboard using computer vision,' in *2010 53rd IEEE International Midwest Symposium on Circuits and Systems*, Aug. 2010, pp. 616–619. DOI: 10.1109/MWSCAS.2010.5548901.

[23] R. Kanchibail, S. Suryaprakash and S. Jagadish, 'Chess Board Recognition,' Spr. 2016.

[24] Y. Xie, G. Tang and W. Hoff, 'Geometry-based populated chessboard recognition,' in *Tenth International Conference on Machine Vision (ICMV 2017)*, vol. 10696, International Society for Optics and Photonics, 13th Apr. 2018, p. 1 069 603. DOI: 10.1117/12.2310081.

[25] Y.-A. Wei, T.-W. Huang, H.-T. Chen and J. Liu, 'Chess recognition from a single depth image,' in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, Jul. 2017, pp. 931–936. DOI: 10.1109/ICME.2017.8019453.

[26] M. A. Czyzewski, A. Laskowski and S. Wasik, 'Chessboard and chess piece recognition with the support of neural networks,' 23rd Jun. 2020. arXiv: 1708.03898 [cs]. [Online]. Available: http://arxiv.org/abs/1708.03898 (visited on 12/09/2020).

[27] T. Romstad, M. Costalba and J. Kiiski, *Stockfish*, official-stockfish, 23rd Sep. 2020. [Online]. Available: https://github.com/official-stockfish/Stockfish (visited on 23/09/2020).

[28] M. Acher and F. Esnault, 'Large-scale Analysis of Chess Games with Chess Engines: A Preliminary Report,' 28th Apr. 2016. arXiv: 1607.04186 [cs]. [Online]. Available: http://arxiv.org/abs/1607.04186 (visited on 23/09/2020).

[29] A. Mehta and H. Mehta, 'Augmented Reality Chess Analyzer (ARChessAnalyzer): In-Device Inference of Physical Chess Game Positions through Board Segmentation and Piece Recognition using Convolutional Neural Network,' *Journal of Emerging Investigators*, 17th Jul. 2020. arXiv: 2009.01649. [Online]. Available: http://arxiv.org/abs/2009.01649 (visited on 20/09/2020).

[30] A. Krizhevsky, I. Sutskever and G. E. Hinton, 'ImageNet classification with deep convolutional neural networks,' *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 24th May 2017, ISSN: 0001-0782. DOI: 10.1145/3065386.

[31] M. A. Czyzewski, A. Laskowski and S. Wasik, 'LATCHESS21: Dataset of damaged chessboard lattice points (chessboard features) used to train LAPS detector (grayscale/21x21px),' 2018. DOI: 10.18150/REPOD.7606646.

[32] J. Hou, 'Chessman Position Recognition Using Artificial Neural Networks.'