Probabilistic Location of a Populated Chessboard Using Computer Vision

Jason E. Neufeld and Tyson S. Hall
School of Computing
Southern Adventist University
Collegedale, TN 37315–0370
Email: {jasonn,tyson}@southern.edu

***Abstract*—Development of autonomic chess-playing robots creates several interesting computer vision problems, including plane calibration and object recognition. Various solutions have been attempted, but most either require a modified chess set or place unreasonable constraints on board conditions and camera angles. A more general solution uses computer vision to automatically determine arbitrary chessboard location and identify chessmen on a standard, unmodified chess set. Although much work has been devoted to probabilistic image recognition in general, this paper presents a novel solution to the specific chessboard location problem that is accurate, less restrictive, and relatively time efficient.**

***Index Terms*—Machine vision, Robot vision systems, Games, Object recognition, Chess.**

I. Introduction

One of many ways that humans and robots can interact is through physical gameplay, using board games such as chess. Autonomic chess-playing robots must have a mechanism to both understand and respond to moves made in-game. Move decisions should be made without unrealistic breaks in the flow of gameplay and with as little manual input as possible. Current implementations vary greatly as to the level of interactivity, as well as which parameters and constraints must be in place [1]–[5].

One of the first hurdles to overcome in reaching higher interactivity is the initial visual detection of the chessboard. Most existing solutions place significant limitations on camera angles and board setup, which are detrimental to the interactive experience and unrealistic outside of the laboratory. This paper shows that by using basic line detection combined with probabilistic reasoning, a computer vision system can locate a chessboard through a single camera at a perspective angle similar to the viewpoint of a human player.

II. Comparison of Approaches

Based on a background survey of existing solutions, there are three major approaches to which primary research has been oriented in chessboard recognition: differential imaging, modified board/pieces, and object recognition. A body of research already exists for the first two methods (see [6] for a good introduction to various chess recognition techniques). However, further research into the third approach may yield major improvements to robotic chess. Table I summarizes the benefits and drawbacks of each approach.

A. Differential Image

The most demonstrated approach to robotic chess using computer vision is differential imaging. This method uses simple vision techniques to discern moves on the board [7]. Before and after images are used to identify board changes, allowing a move to be deduced with relative certainty [1], [8].

The first problem that must be overcome with this method is board position. Generally, a camera position directly above and orthogonal to the board is ideal for this approach, as it is the simplest in terms of image processing. However, this sacrifices flexibility [9]. To determine board position, a simple Hough transform [10] is the most frequently utilized method. Although the Hough transform generally gives a high degree of accuracy, Tam, et al., have proposed a method of board detection that nearly eliminates false positives and misses from grid detection by combining line-based grid detection with domain knowledge of the chessboard [9].

Next, pieces must be found on the board. The most common approach is to locate the areas that are statistically the most populated on the grid. Cour, et al. solve this by using the average intensities in a square [1]. Depending on whether the square is dark or light, a different average threshold is used to determine whether a black piece or white piece is located on that square. Problems include gradients of brightness, bright dark regions, and parasite spots. This method, then, is particularly prone to changes in lighting.

The primary drawback of using the differential imaging approach is the lack of individual piece recognition. Because of this, a game that started in an arbitrary position would require manual entry of piece locations into the chess engine. The vision system could also be thwarted as simply as swapping chessmen, and the physical setup is more constrained, especially when an orthogonal camera view is required.

B. Modified Board/Pieces

Most consumer electronic chess boards, such as those marketed by Excalibur [2] and Saitek [3], use board or piece modification. From simple magnetic pieces to more advanced individual piece sensors, this is an attractive method because its detection methods are more fool-proof. Because it uses non-visual detection methods, it obviates the need for a camera setup or computer vision of any kind. Some implementations can also be easily transported as a self-contained unit.

978-1-4244-7773-9/10/$26.00 ©2010 IEEE

Method	Advantages	Disadvantages
Differential Image	• Simplest implementation • Works with almost any chess set	• Cannot recognize individual pieces, • Can be easily thwarted under certain conditions • More susceptible to lighting variations
Modified Board/Pieces	• Individual piece recognition is possible • Electromagnetic boards can automatically move pieces • Minimal computational complexity • Most self-contained	• Most expensive • Requires changes to boards and pieces • Does not allow the use of multiple chess sets
Object Recognition	• Recognizes individual pieces • Can be used with arbitrary board configurations • Can support abstract chess sets	• Most complex implementation • Requires detailed internal representations of chess pieces

TABLE I
SUMMARY OF APPROACHES TO ROBOTIC CHESS

Drawbacks include expense and engineering requirements of such a board. Since the board itself and individual pieces must be altered, replacing the board or pieces is non-trivial. Also, although some boards have effective electromagnetic motion that obviate the need for a robotic arm, others require the user to move pieces manually. Novag's 2Robot electronic chess set solves this by integrating a robotic arm into their consumer chess set [4].

One effective implementation of this method, called Robochess, was created by Ebrahim Jahandar [5]. Magnets are implanted in each of the chess pieces, and occupied squares are determined via magnetic force. However, this set does require the pieces to be configured in prescribed locations initially as it does not recognize individual piece identities.

C. Object Recognition

The object recognition method shares some commonality with the differential image approach, especially in the areas of board detection and electromechanical motion. Instead of using differentiation and a board in memory to detect moves, however, an object recognition approach would be able to look at any arbitrary configuration of chessmen and present the board layout to the chess engine.

Although the robustness of this approach seems to solve most of the existing problems in robotic chess, its implementation presents a few non-trivial challenges. Most apparent is the complexity involved in piece detection. Object recognition techniques must be employed to identify features unique to each piece, and new features would need to be defined if a different style of chessmen was used. Also, since the camera angle must be at a perspective angle to be able to visually recognize the identities of the pieces, the option of having a simple top-down view with trivial grid detection is no longer available. Therefore, a more sophisticated board locating algorithm must be used.

The advantages appear to outweigh the disadvantages of this approach. Not having to rely on a particular angle simplifies camera mounting and setup. Also, research in this area could be applicable to other object recognition problems. There is a small amount existing research on the topic. Yali Amit explores distinguishing an individual piece based on reference feature points around its outline [11]. Herbin proposes a recognition system of chess pieces using a moving camera, which could be useful if the parameters of a particular implementation allowed for a moving board [12].

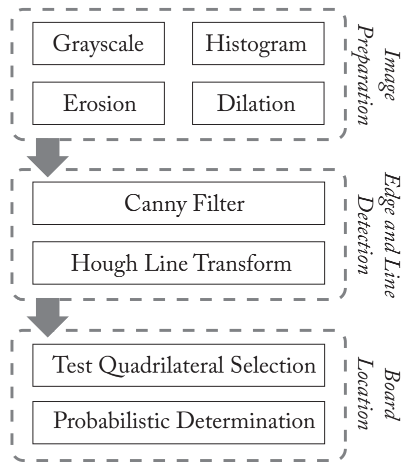
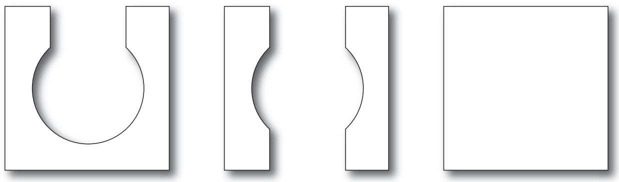


Fig. 1. Major steps used in the presented board detection algorithm.

Despite this data, research on integrating piece recognition with a complete board setup is sparse. Tam, et al. have indicated that this is their next area of research in [9], but as of writing no complete implementations of this method have been found.

III. "HIGHEST PROBABILISTIC MATCH" METHOD FOR BOARD DETECTION

As summarized in Fig. 1, a board detection method has been developed to allow for quick board recognition and to easily integrate with a future piece detection system. This method uses a line transform to detect potential board edges in the image. Then, using intersections between horizontal and vertical lines in the four quadrants of the image, it creates a list of potential quadrilaterals. Each quadrilateral is then tested for the probability that it matches a chessboard pattern. The quadrilateral that has the highest probability ranking at the end of the test is returned.



(a) (b) (c)

Fig. 2. Piece masks were used to increase accuracy of board detection by avoiding sampling from potential chessmen on board squares. (a) Mask 0: Center and top area obscured from test. (b) Mask 1: Center, top, and bottom areas obscured. (c) Mask 2: Control mask, with nothing obscured.

A. Initial Image Preparation and Line Detection

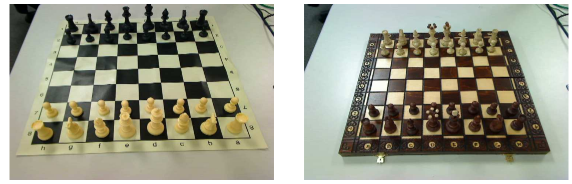
A few basic filters are applied to the image in order to make the board detection process more efficient and reduce noise. A grayscale copy of the input image is used for both line detection and final board comparison. The image histogram is equalized to normalize brightness and increase contrast. Finally, erosion and dilation filters are applied in succession to smooth out noise in the image and to reduce unnecessary lines from being detected.

The filtered image is passed through an edge transform to prepare it for line detection. The Canny method has a high signal-to-noise ratio [13], making it ideal for this study. The standard Hough line transform [10] is used to detect the lines, which are then sorted separately into groups of horizontal and vertical lines. Quadrilaterals are found by 1) matching a horizontal and a vertical line that intersect in the upper left quadrant, 2) selecting a vertical line that intersects with the same horizontal line in the upper right quadrant, and 3) selecting a horizontal line that intersects with the last vertical line in the lower right quadrant. The lower left intersection is found using the first vertical line and the last horizontal line, yielding a time complexity of $O(n^2)$, where n is the number of lines.

An alternative method to finding quadrilateral methods is corner detection; however, this would have a significant negative impact on performance, because the corners returned would have no information as to which points are co-linear with other points. Since the algorithm presented in this paper uses intersections of lines to determine corner points, the total test count is significantly lower.

B. Board Probability Match

Deriving the board probability is relatively straightforward. For each quadrilateral, a square chessboard shape is projected via a perspective transform onto the test quadrilateral. Each square, now in perspective, is then sampled for the average intensity value of its pixels. Because a contrasting chessman may populate a square, a mask is used to sample around where a piece would likely be, as shown in Fig. 2. The differences of the expected dark squares and light squares against their respective lowest and highest possible intensity values determine the ultimate board probability. The lower the average differences, the higher the probability that the quadrilateral in question is a real chessboard.



(a) (b)

Fig. 3. Two boards were used to obtain test images. (a) Board 1: A regulation, tournament chessboard. (b) Board 2: An ornate wooden chessboard.

C. Optimizations

A few base constraints have been defined to improve execution time. First, the orientation of the board is assumed to be standard, with a dark square in the lower left, which cuts execution time in half as potential boards do not need to be tested against both rotations. Also, the board detector does a running probability check for each quadrilateral. Since the high contrast of actual chess squares will show high probability values after only a few squares are tested, the current quadrilateral being tested can be pruned if the running probability falls below a set threshold.

IV. Testing and Results

A total of 153 images were taken at various angles, piece setups, and lighting conditions. An autofocusing webcam was used to capture images at NTSC standard resolution (720x480). The two boards shown in Fig. 3 were tested. The first is a regulation tournament-style chess board. The checker pattern on the board is highly contrasted with a well-defined border, low reflectivity, and minimal decoration that would contribute to interference. It is large enough that pieces can be visually recognized easily. Also, since it is a tournament board, it is a fair test for the legitimacy of this algorithm. The second board is an ornamented wooden set. It has a higher reflectivity and lower contrast for line detection. Its purpose for selection was as a stress test for the board detector. An Apple MacBook Pro with a 2.8GHz Intel Core 2 Duo processor was used for evaluation.

Testing metrics were the accuracy of the test and total execution time. The results were visually inspected and marked as an accurate match, a false positive, or no match at all. As Fig. 4 shows, the two obscuring masks, Mask 0 and Mask 1, both had the same accuracy rate with the first board at 91.6% of test boards correctly identified with no false positives. Mask 2 only achieved a 72.9% detection rate with no false positives. As expected, the detection rate for the ornamented board was quite low, which can be attributed to the difficulty of the line transform correctly detecting all four edge lines, especially with a low-contrast board. Surprisingly, Mask 2 had almost twice the detection rate (19.6% vs. 10.9%) of Masks 0 and 1. The masks had false positive rates on the second board of 28.3%, 23.9%, and 17.4%, respectively.

Board location times showed that Masks 0 and 1 performed significantly better than Mask 2, taking on average 26.2%

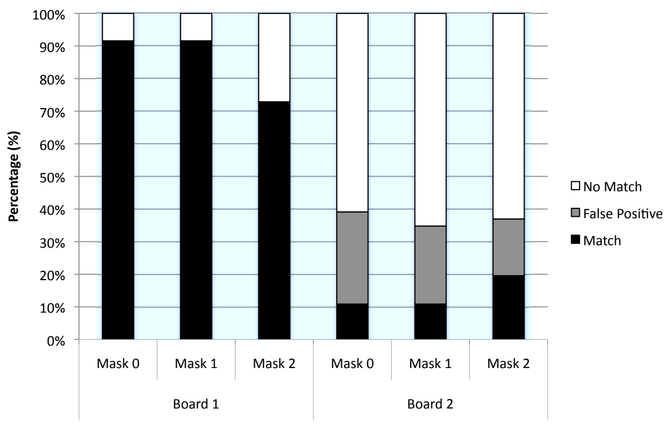


Fig. 4. Percentage of test image matches for all piece masks and both boards.

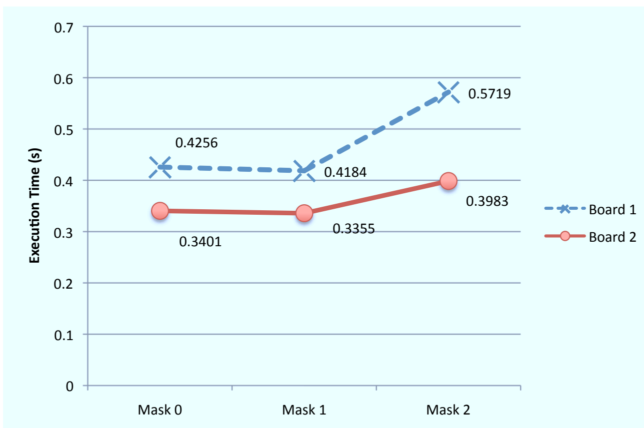


Fig. 5. Average board location times for each board and piece mask.

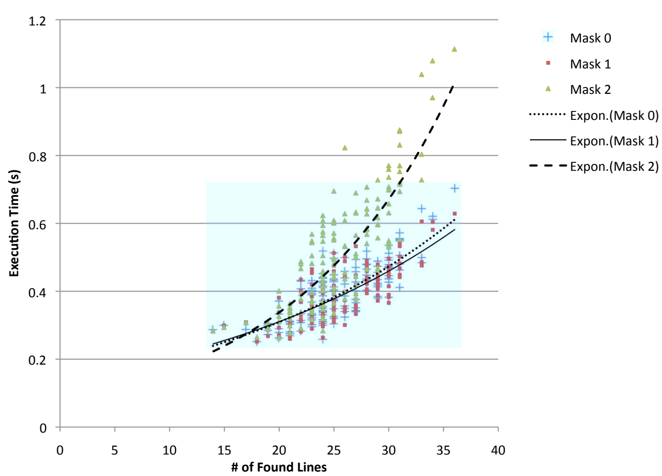


Fig. 6. Number of lines found *(n)* vs. execution time.

and 15.2% less time on boards 1 and 2, respectively. Mask 1 was slightly faster than Mask 0 (see Fig. 5). Comparison of the amount of lines found in each image vs. board location time gave similar results, with Mask 2 showing a significantly steeper increase in location time for greater line amounts than Masks 0 or 1, with Mask 1 slightly edging out Mask 0 as the fastest piece mask (see Fig. 6).

V. Conclusion

The results from the board analysis method presented are promising. It has been shown that a board with reasonable constraints can be probabilistically detected relatively quickly. This method, combined with a chess AI and robotics, could ultimately create a realistic chess playing robot that does not require extensive board preparation or costly hardware.

Improvements to this method would include more robust line detection methods to increase relevant edge detection while compensating for lower thresholds by heuristically pruning noise lines. This would further improve execution time, as well as allow more ornate boards to be detected with higher consistency. Also, testing this algorithm against a live video feed would provide valuable performance and accuracy data.

Future research will be directed towards individual piece recognition, for which this study has laid the groundwork. One possible method of piece recognition involves a combination of computer vision, computer graphics, and probabilistic matching. 3D renderings of individual pieces would be overlaid on possible board locations, and probabilistic comparison would be used to detect the likelihood of that piece being present. Since the perspective of the board plane is known and the amount of test spaces are finite, this proposed method shows potential towards solving the second major problem of robotic chess recognition.

References

[1] T. Cour, R. Lauranson, and M. Vachette, "Autonomous chess-playing robot," 2002, undergraduate thesis, Ecole Polytechnique.

[2] Excalibur Electronics. Chess and electronic chess. [Online]. Available: http://www.excaliburelectronics.com/catgry-chess.html

[3] Saitek. Computer chess games. [Online]. Available: http://www.saitek.com/uk/prod/chess.htm

[4] P. T. Limited. 2robot - a new generation of chess computers has arrived! [Online]. Available: http://www.perfectchessbynovag.com

[5] E. Jahandar. Chess playing robot (robochess). [Online]. Available: http://www.jahandar.ir/Chess-Playing-Robot

[6] S. Blunsden, "Chess recognition," Ph.D. dissertation, University of Plymouth, 2003, BSc (Hons) Computing and Informatics.

[7] F. Groen, G. der Boer, A. van Inge, and R. Stam, "A chess playing robot: lab course in robot sensor integration," *Instrumentation and Measurement Technology Conference, 1002. IMTC '92., 9th IEEE*, no. 1002, May 1992.

[8] E. Sokic and M. Ahic-Djokic, "Simple computer vision system for chess playing robot manipulator as a project-based learning example," *IEEE Symposium on Signal Processing and Information Technology (ISSPIT 2008)*, December 2008.

[9] K. Tam, J. Lay, and D. Levy, "Automatic grid segmentation of populated chessboard taken at a lower angle view," *Computing Techniques and Applications, 2008. DICTA '08. Digital Image*, pp. 294–299, December 2008.

[10] R. Duda and P. Hart, "Use of the hough transform to detect lines and curves in pictures," *Comm. ACM*, vol. 15, no. 1, January 1972.

[11] Y. Amit, *2D Object Detection and Recognition*. Cambridge, MA: MIT Press, 2002, p. 216.

[12] S. Herbin, "Recognizing 3d objects by generating random actions," *Computer Vision and Pattern Recognition Proceedings CVPR '96, 1996 IEEE Computer Society Conerence on*, pp. 35–40, 18–20, June 1996.

[13] M. Sharifi, M. Fathy, and M. T. Mahmoudi, "A classified and comparative study of edge detection algorithms," *International Conference on Information Technology; Coding and Computing (ITCC'02)*, 2002.