International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Chesslogic: AIML-Powered Chess Arbitrator

Dr. Ramalingam H M[1], Prakash Nayak, Tejas[2], Krithik[2], Jasvika[2]

[1]Associate Professor, [2]Student

Dept. of ECE, Mangalore Institute of Technology and Engineering, Moodbidri, India

ABSTRACT:

Chess, a strategic and intellectual board game, has seen a transformative evolution with the integration of digital technologies. This paper delves into the methodologies and techniques used for detecting and analyzing chess moves, focusing on the intersection of computer vision, machine learning, and artificial intelligence. We provide a comprehensive overview of current state-of-the-art methods for move detection, including convolutional neural networks (CNNs), optical character recognition (OCR), and real-time video processing algorithms. Key challenges in chess move detection are discussed, such as varying board designs, fluctuating lighting conditions, and partial occlusions, which can affect the accuracy and reliability of detection systems. We explore solutions to these challenges, highlighting advancements in image processing and machine learning models that improve the robustness and precision of move detection. Furthermore, the paper examines the practical applications of these technologies in various domains. In the realm of competitive chess, move detection systems enhance the integrity and fairness of the game by providing real-time monitoring and analysis. For enthusiasts and players, these technologies offer improved tools for training and coaching, enabling detailed analysis of games and facilitating skill development. In the context of spectatorship, automated move detection enriches the viewing experience by providing instantaneous updates and insights during live broadcasts. The integration of AI in chess is not only preserving the traditional essence of the game but also driving innovation and new research directions in computer vision and pattern recognition. This paper aims to provide a thorough understanding of the current landscape and future potential of chess move detection technologies, emphasizing their significance in the ongoing digital transformation of chess.

1. INTRODUCTION

A game with centuries of history, chess is more than just entertainment. Two thoughts engage in strategic combat on a checkered terrain in this intellectual warfare. But why is chess such a compelling game, and are there any drawbacks to this age-old sport? The secret to chess's charm is its flawless fusion of depth and simplicity. Anyone can pick up the pieces and play since the fundamental rules are simple to understand. Below this easily visible exterior, however, comes a vast array of strategic opportunities. Every action has an impact on everything, necessitating thoughtful preparation and foresight. It takes strong analytical, problem-solving, and critical thinking abilities to become a master chess player. Chess has several advantages that go well beyond the board. It can enhance memory, focus, and spatial thinking, according to studies. It encourages innovation as you devise cunning plans and anticipate your opponent's next move.. In addition to teaching important life lessons, chess also helps players develop their decision-making and resource-management abilities. Chess has benefits, but it also has disadvantages. Strong chess matches can be psychologically exhausting, especially for novices up against more experienced players, and can result in frustration and despair. The game can take a lot of effort to master, which could take time away from other activities. Furthermore, some chess players may experience harmful amounts of stress due to the game's competitive character. These drawbacks can be lessened, though. Less pressure may be applied by approaching the game of chess with an emphasis on fun and education rather than winning alone. Playing with friends who are at the same skill level as you or using online matching services that match players based on skill can guarantee a more balanced and pleasurable experience. To sum up, chess provides a special fusion of strategic depth and intellectual difficulty. This game may provide you a lifetime of intellectual stimulation, sharpen your mind, and enhance your cognitive abilities. However, to completely reap the rewards of chess, one must be aware of any potential negatives and keep a healthy mindset. Chess provides a special fusion of historical context, strategic nuance, and intellectual challenge. It's a voyage that may bring you lifetime intellectual stimulation, sharpen your mind, and enhance your cognitive abilities. Thus, whether you're an experienced tactician or an inquisitive novice, think about stepping into the checkered battlefield and starting your own chess journey.

2. LITERATURE SURVEY

S. Maharaj, N. Polson, and A. Turk.,[1] Stockfish and LC Zero are two competing chess engines. Stockfish's search algorithm was more efficient in solving Plaskett's Puzzle, searching through 1.9 billion positions. LC Zero's selective search was less efficient due to wrong lines and its evaluation function failed to recognize the positional potential of knight sacrifices. Machines can potentially possess a similar imagination, as they can predict likely checkmate positions and use these to condition the search. This process extends the assessment of win probability by conditioning on the event of reaching a certain checkmate position. However, it's crucial to avoid "magical thinking" and ensure that all relevant lines are examined to verify a checkmate as a forced win.

H. Vasconcelos et al., [2] The study supports the claim that as tasks increase in difficulty, overreliance on AI's predictions increases. This is due to the cost-benefit framework, as increasing task difficulty increases the cost of engaging with the task. However, the strategy to rely on AI does not incur the same costs, as people are not engaging with the task's difficulty. The study also found that overreliance increases with explanations in hard tasks, which fits into the cost-benefit framework, as explanations in hard tasks reduce the cost of doing a task properly.

Kalpesh H. Zurange et al.,[3] The H-Bot concept combines two rotary drives connected by a single H-shaped circumferential timing belt around two staggered linear axes in a gantry-like configuration. This mechanism allows for high dynamic values and a more manageable working envelope, compared to delta-robot-like kinematics. The reed switch matrix, which requires 64 reed switches, can cause "ghost" activations, which can be solved by adding a diode in series with each switch.

Reid McIlroy-Young, Siddhartha Sen, Jon Kleinberg, Ashton Anderson et al.,[4] The study uses Stockfish and Leela chess engines to analyse human moves and mistakes. Stockfish uses heuristic functions and classical alpha-beta game tree research to evaluate positions, while Leela uses a deep neural network and Monte Carlo tree search for position estimation. The researchers repurpose Leela's neural network architecture for move matching tasks, learning from real human games played on Lichess.org, a popular, free, open-source chess platform with over 1 billion games played daily. The games database is composed of genuine human games, with players playing at different time controls and having separate ratings for each format.

Rohit Verma, Sanchit Verma et al.,[5] Testing work is a crucial module in chess engines, with the main improvement coming from transforming the test function. The input is a chess area, and the output is a number. A different data algorithm mimics a grand mastermind, which is fundamental for understanding chess and chess engines. Stockfish uses methods to calculate evaluation, including Material (material value) and Pawn Structure (pawn structure). Pawn structure is determined by the position of pawns on the board, such as doubled pawns or passed pawns. Evaluation of pieces is influenced by board state, with each piece having specific rules to make its value better or worse. Evaluation patterns require additional knowledge for certain positions, such as when a bishop is fianchettoed for a knight or bishop.

D. A. Christie, T. M. Kusuma, and P. Musa et al.,[6] Chess cell detection is a method that searches the entire 8x8 cells in a chessboard to give an identity to each cell. This method includes preprocessing, line detection, line selection, line intersection, and cell identification. Cell identification involves processing intersection points to create correct components-forming cells The total number of points used is 72 instead of 81, as the intersection points at the last row are represented by the points above it which is used to detect the borders of the boxes in chess board.After the coversion of raw image into gray scale both clours are set to certain threshold frequency which is identified bold frequency which is identified by the engine to know the difference between two types of cells.

Magesh Surya Ambikapathi et al.,[7] Chess engines, designed for computer use, have evolved over time to improve performance. Some games are designed for network play, allowing players to compete against each other or the computer. The Min-Max Algorithm is a decision-based method used in game trees to minimize worst-case losses. Pruning is a process of cutting off the branch when the lowest or maximum node has reached the required value. Surakarta Chess is a Chinese variant with six horizontal and vertical lines, while Hexagonal Chess has 91 cells and symmetrical rules.

Xiaoyu Zhang, Wei Liu, Linhui Lou, Fangchun Yang et al.,[8] A visualization system framework was designed, divided into three parts: web front-end server, web back-end server, and GPU server. The user accesses the visual page, enters their bidding process, and sends tasks to the web back-end server. The GPU node processes the task, stores the prediction result, and sends it to the front-end server. The web front-end provides a user interface and displays the results. The web back-end server interacts with the GPU server, performs user input verification, and has an error handling mechanism.

Francisco S. Marcondes et al.,[9] It assesses the element's favourable and contrary evidence, balancing them within a lattice. In this context, V (true) and F (false) are binary models, and the evidence is a model prediction for each class. Paraconsistent analysis can address both uncertainty and unknown, providing a continuous uncertainty degree connecting lattice extremities. The open world recognition problem is intrinsically problematic due to the infinity property of open sets. Some attempts at handling this problem range from the use of a threshold and a "garbage" class to the open-max approach.

Azlan Iqbal et al.,[10] The new variant of SSCC introduces a new rule allowing players to switch armies with their opponent when a chain is formed on the board by the last piece, enclosing at least two empty adjacent squares. This introduces levels of complexity beyond the standard game without the computational burden of a larger search space. The new variant poses new challenges to gaming AI without the computational burden of a larger search space. Concepts such as intuition and perception are critical to high-level play, and further work in the area would benefit from a creative, mathematical demonstration of the increased strategic complexity of SSCC.

3. AIML

Chatbots and virtual assistants are created using AIML (Artificial Intelligence Markup Language), an XML-based programming language. It works by establishing guidelines for conversation. These rules establish chatbot responses based on user input, much like a script. Pattern matching is a technique used by AIML to find applicable rules when a user types. The relevant automatic response is then retrieved by the programming for the chatbot to provide. Although useful, AIML has a rule-based approach, which restricts its ability to respond to preprogrammed scenarios and makes it more difficult to comprehend the complexity of real language than some more sophisticated AI models. Although AIML has a place, chatbot creation is a constantly evolving field. More sophisticated AI methods, such as natural language processing and machine learning, are being used to develop chatbots that have a deeper comprehension of language.

History : In the late 1990s, AIML (Artificial Intelligence Markup Language) was developed by Wallace Richard. Its goal was to make chatbot creation easier by offering a rule-based methodology. Consider a dialogue script; AIML enables you to establish these guidelines, dictating the chatbot's replies. according to user input. When someone types something, it matches patterns to determine which fit is optimal. AIML became well-known due to its open-source and user-friendly design. But compared to more sophisticated AI models, its comprehension of natural language is limited by its dependence on pre-programmed rules. Even though it's still utilized in certain chatbots, AIML has opened the door for more advanced chatbot building methods.

4. FLOW CHART

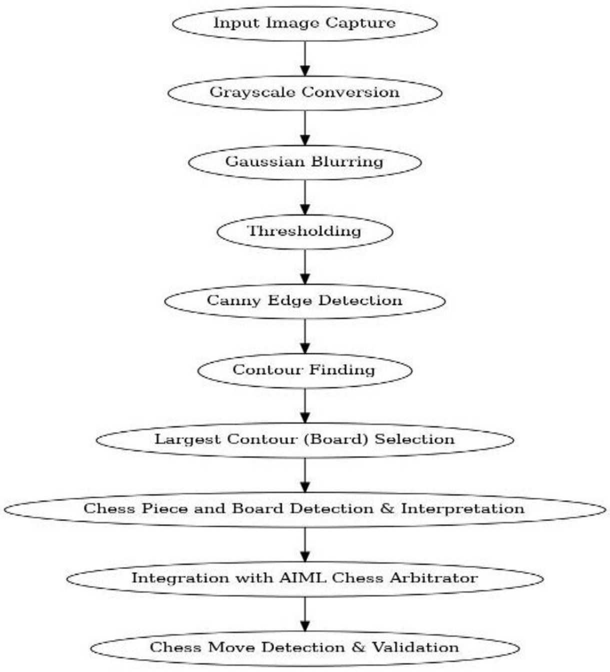

Figure 4.1: Flow Chart

The block diagram you sent shows the steps involved in an algorithm for chessboard and chess piece detection and interpretation.

Here's a breakdown of the steps:

4.1 Input Image Capture: The first step is to capture an image of a chessboard. This could be done with a digital camera or webcam.

4.2 Grayscale Conversion: The image is then converted from colour to grayscale. This simplifies the image processing steps that follow.

4.3 Gaussian Blurring: A Gaussian blur is applied to the image. This helps to smooth out noise and make it easier to detect the edges of the chessboard and chess pieces.

4.4 Thresholding: The image is then threshold. This means that each pixel in the image is converted to either black or white based on a certain threshold value. Pixels that are lighter than the threshold are converted to white, and pixels that are darker than the threshold are converted to black.

4.5 Canny Edge Detection: Canny edge detection is applied to the image. This is an algorithm that is specifically designed to detect the edges of objects in images.

4.6 Contour Finding: The algorithm then finds the contours in the image. A contour is a closed loop of connected edge pixels. The contours in the image will correspond to the edges of the chessboard and chess pieces.

4.7 Largest Contour (Board) Selection: The largest contour in the image is likely to be the chessboard. This contour is selected and used to define the region of interest for the next step.

4.8 Chess Piece and Board Detection & Interpretation: Here, the algorithm focuses on the defined region of interest to detect and interpret the chess pieces and board. This likely involves analyzing the contours and applying chessboard knowledge to identify the squares and their corresponding chess pieces.

4.9 Integration with AIML Chess Arbitrator: The final step involves integrating the chess piece and board detection with an AIML chess arbitrator. AIML (Artificial Intelligence Markup Language) is a rule-based language that can be used to create chatbots. In this case, the AIML chess arbitrator is likely a software component that uses the information about the chessboard.

5. METHODOLOGY

5.1 Requirements Gathering:

Defining the specific requirements of the AI-based arbitrator, considering the rules and regulations of chess tournaments, common disputes, and the need for real-time decision-making. Determining the goals and parameters of an AI-powered chess arbitration system is essential for obtaining information, as it clarifies whether the system will manage move validation, rule enforcement, or dispute settlement. Understanding the requirements and expectations of important stakeholders—such as chess players, tournament directors, and arbiters—begins with identifying them. Non-functional criteria should deal with performance, accuracy, scalability, security, and usability, while functional needs should cover the system's capacity to validate moves, enforce rules, settle disputes, and offer game analysis. The sorts of input data that are required, such as moves and game state, as well as the training data for the AI model, are specified in the data requirements. Selecting the right algorithms and preparing training data are important factors to take into account when using AI and machine learning. The user interface ought to be simple to use.

5.2 Data Collection and Preprocessing:

Preprocess the data to extract relevant features, such as board positions, move sequences, and historical game outcomes. In order to guarantee that an AI-based chess arbitration system functions properly, data collecting entails obtaining a range of essential facts. The AI needs historical game data, which includes previous moves, game states, and results from several chess games, in order to be trained. Move data guarantees that moves follow the rules of chess, whereas real-time game state data is required to verify current positions and moves. Furthermore, gathering disagreement data aids in the AI's ability to manage and settle disputes by drawing on earlier decisions. Player data offers insights into performance and conduct, and rule data from official chess rulebooks guarantees adherence to rules. Annotated game examples are included in the training data, which improves the AI's accuracy and decision-making. User feedback is essential for system improvement, and contextual data

5.3 Model Selection:

Choosing an appropriate AI model for chess arbitration, such as a deep neural network, considering the complexity and hierarchical nature of chess gameplay.

For AI-based chess arbitration, the right mode must be chosen by taking into account the needs and objectives of the system. Rule-based systems, which use pre-established chess rules, are straightforward and simple to use, but they might not be flexible enough in complex scenarios. By using previous data to learn, machine learning models can adjust to a variety of situations and provide sophisticated arbitration, but they do require a large amount of training data. The advantages of rule-based and machine learning techniques are combined in hybrid systems, which trade off robustness and adaptability but add complexity to integration. Expert systems utilize a database of chess knowledge to make intricate judgments, whereas decision trees provide an organized, albeit perhaps laborious, method. While they require a large amount of processing power, neural networks are capable of recognizing intricate patterns and producing precise predictions. Reinforcement learning using incentives and sanctions

5.4 Training the Model:

Training the selected model using the annotated dataset, employing techniques like transfer learning or fine-tuning to leverage existing chess knowledge.

To guarantee accuracy and dependability, an AI model for chess arbitration must be trained via a methodical process. Data collection, including acquiring dispute records, past game data, and chess rules, is the first step in the process. To guarantee consistency and relevance, the data must be cleansed, annotated, and standardized after it has been gathered. After that comes feature selection, when important aspects are engineered and recognized, like board configuration and move history. Selecting the appropriate algorithms is essential; based on the particular requirements, choices can include decision trees, neural networks, or reinforcement learning. After then, the model is trained on the prepared data, carefully dividing it into test, validation, and training sets to maximize efficiency and prevent overfitting. Evaluation criteria like F1 score, recall, accuracy, and precision are utilized

5.5 Real-time Decision Making:

Implementing a real-time decision-making system that analyzes the current game state, identifies disputes, and provides timely and accurate rulings.

Testing and Validation: Conducting extensive testing under various scenarios, including different levels of play, rule violations, and ambiguous situations.

5.6 Feedback Loop and Continuous Improvement:

Establishing a feedback loop to collect information on the AI arbitrator's performance from users.

5.7 Ethical Considerations:

Addressing ethical concerns related to AI-based decision-making, including transparency, accountability, and potential biases.

Implementing safeguards to ensure the system operates ethically and avoids unintended consequences

6. BOARD DETECTION

6.1 Image Preprocessing:

Grayscale Conversion: Converting the image from colour to grayscale simplifies processing and reduces computational resources during pre-processing.

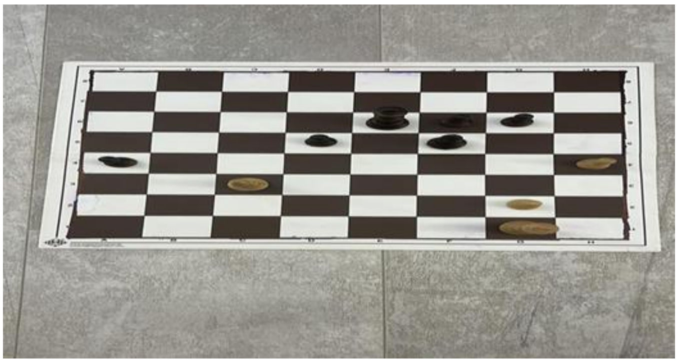

Figure 6.1: Before Detection

6.2 Edge Detection:

Canny Edge Detection: This is a popular algorithm for finding edges in images. It effectively identifies significant changes in pixel intensity, highlighting the boundaries of the chessboard and pieces.

6.3 Line Detection: Hough Transform:

This technique is often used to identify lines in an image. In the context of a chessboard, it helps locate the straight lines formed by the grid of squares.

Figure 6.2:After detection

6.4 Contour Analysis:

Contour Finding: Algorithms identify connected edge pixels that form closed loops (contours). Contours corresponding to the chessboard will be prominent due to its well-defined borders. o Shape Analysis: Analyzing the shape properties (e.g., number of corners, aspect ratio) of the detected contours helps distinguish the rectangular chessboard from other elements in the image.

6.5 Geometric Verification:

Grid Verification: Since a chessboard has an 8x8 grid, the algorithm might check for the presence of eight horizontal and eight vertical lines (or a close approximation) to confirm the detection.

6.6 Machine Learning:

Deep learning models trained on chessboard images can achieve high accuracy in detection, especially under challenging conditions like lighting variations or tilted angles.

7. PIECE RECOGNITION

The heart of the project lies in piece recognition, which involves using a pre-trained machine learning model provided by Roboflow. This model has demonstrated an impressive Mean Average Precision (mAP) of **80.2%**, showcasing its ability to accurately identify chess pieces. The model achieves a remarkable precision of **93.0%** and a recall rate of **75.0%**, making it a reliable choice for classifying chess pieces. Gaussian blurring is a technique used in image processing to soften an image. Imagine blurring a photograph with a translucent sheet. It works by applying a mathematical function (the Gaussian function) to each pixel, considering the surrounding pixels. This reduces sharp contrasts and creates a smoother appearance, often used for noise reduction or artistic effects. The strength of the blur is controlled by a parameter sigma, where a higher sigma results in a more blurred image. Chess piece recognition involves identifying the type (king, queen, rook, etc.) and position of each piece on the chessboard using computer vision.

Figure 7.1: Blurred image before piece detection

Figure 7.2:Image after piece recognition

Here's the gist:

Preprocess: The image is converted to grayscale and blurred to reduce noise.

Segmentation: Techniques isolate individual pieces on the board. This might involve edge detection and contour analysis.

Template Matching or Machine Learning: Pre-defined templates representing each chess piece (or a machine learning model trained on labeled chess piece images) are used to compare and identify the type of piece based on its shape and position.

Positional Verification: The detected piece's location on the board is verified to ensure it aligns with the chessboard grid and standard piece placement rules.

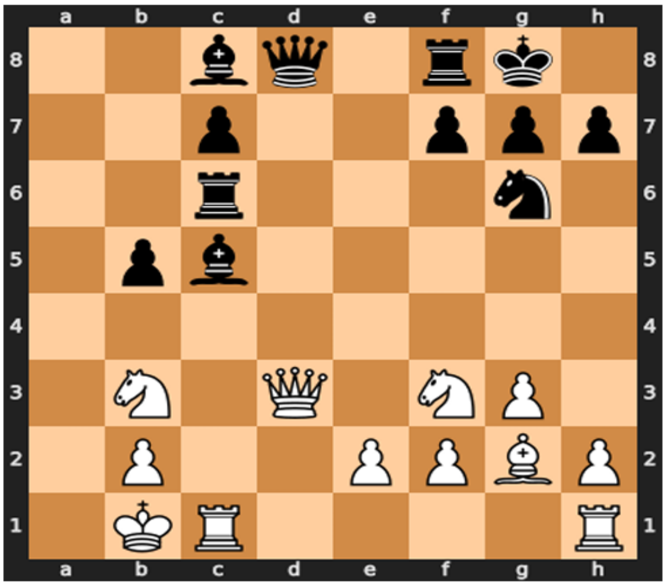

Figure 7.3: Digitally converted image

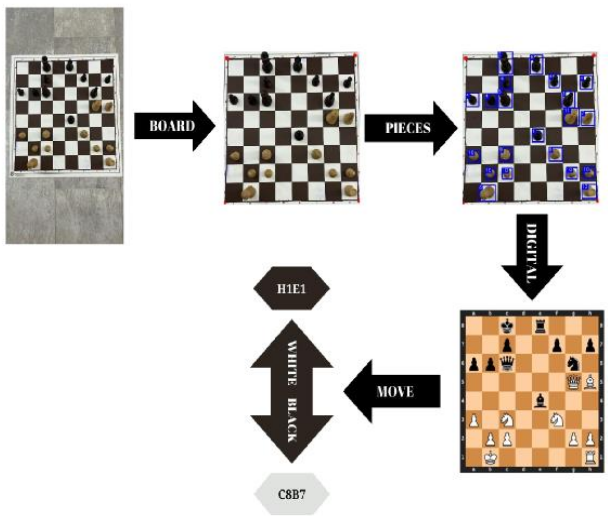

Figure 7.4: Step by step process of project

8. APPLICATIONS

The technology you're working on to recognize chessboards and chess pieces opens up fascinating possibilities beyond merely playing the game. Envision creating an AI that is proficient in chess analysis, move planning, and competitiveplay. Tools for analyzing chess could also undergo a revolution. After a game, players can use your program to take a photo of the chessboard, assess positions, and pinpoint areas that need work. By merely setting up the board and allowing your program to provide feedback, students can practice tactics or analyze well-known games in educational chess programs. Players who are blind or visually challenged have additional options. Your idea could increase everyone's ability to play chess by transforming the visual clues of the board and pieces into audible or tactile signals. Live chess broadcasts might also profit from this. Standard chess notation might be generated by automatically recognizing the board and pieces, reducing manual labor and improving live commentary. These are but a handful of the many uses that demonstrate the enormous potential of your initiative in the chess community and beyond.

9. SOFTWARE COMPONENTS

Anaconda promt: Anaconda Prompt is a command-line interface that is included with the popular Python and R programming languages distribution Anaconda, which is used for data research and scientific computing. It enables users to run commands pertaining to Python and other supported languages, as well as manage environments and packages.

By looking for "Anaconda Prompt" in your operating system's application search or launcher, you can access Anaconda Prompt on your computer. Once it's opened, you may use it to manage virtual environments, install packages using pip or conda, run Python scripts, and carry out different system tasks associated with your Anaconda installation. Please don't hesitate to ask questions or submit specific tasks pertaining to the Anaconda Prompt! In addition to environment management and packaging, it offers standard shell commands for file operations, directory navigation, and Python script execution. Through its integration with Anaconda Distribution, a large library of pre-built tools and packages for computational and data analytic activities is made available. Anaconda Prompt is a crucial tool for developers and data scientists looking to improve workflow speed and control because it also lets users manage routes, customize their Python environment, and adjust system settings.

10. LIMITATIONS

10.1.Image Quality Dependence: Image quality can have a big impact on accuracy. Inadequate illumination, shadows, limited resolution, or skewed angles can all make it difficult for the algorithms to identify the board and pieces with accuracy.

10.2. Occlusion Issues: Errors in recognition may arise from pieces overlapping one another or from items on the board. The technology may have trouble determining the kind or location of obscured parts.

10.3. Non-Standard Chessboards: The chessboards with regular colors and patterns may be used to instruct the technology. Distinctive boards using distinct colors or designs may cause problems with recognition.

10.4. Computational Cost: Especially when they involve machine learning, complex algorithms might demand a lot of computing power.

10.5. Limited Context Understanding: The primary focus of current technology is on identifying individual components and the status of the board. It's still difficult to comprehend the context of the game, such as possible future moves or strategic ramifications.

11. CONCLUSIONS

Chessboard and chess piece recognition technology unlocks a world of possibilities. It paves the way for intelligent chess-playing AIs, interactive learning tools, and wider accessibility for visually impaired players. However, challenges remain. Image quality, occlusions, non-standard boards, and computational limitations can hinder accuracy. Despite these limitations, the technology offers a valuable foundation for the future of chess analysis, education, and AI development. As research progresses, overcoming these limitations will lead to even more robust and versatile chess recognition systems, pushing the boundaries of the game we love.

12. SCOPE

Advanced Image Processing: Improvements in image processing techniques like noise reduction and handling variations in lighting and angles will ensure better recognition under diverse playing conditions.

Deeper Learning Models: Utilizing more powerful deep learning models trained on vast datasets of chessboard images will enhance the ability to handle occlusions, non-standard board designs, and complex piece arrangements.

3D Reconstruction: Integrating 3D reconstruction techniques could allow for recognition even from tilted angles or in situations where pieces are stacked on top of each other.

Real-time Processing: Advancements in hardware and software optimization will enable real time processing on mobile devices and embedded systems, facilitating wider application in areas like live chess broadcasting and mobile chess analysis tools.

Integration with Chess Logic Engines: Combining recognition technology with powerful chess logic engines will enable the creation of AI opponents that not only understand the board state but can also strategize and make optimal moves, pushing the boundaries of chess-playing AI.

Multimodal Interaction: Future systems might incorporate voice commands or touch interfaces for seamless interaction with the chessboard and analysis tools, making chess more accessible for users with varying needs and preferences.

13. REFERENCES

[1] S. Maharaj, N. Polson, and A. Turk, “chess AI: Competing Paradigms for Machine Intelligence,” 2022.

[2] H. Vasconcelos et al., “Explanations can reduce overreliance on AI systems during decision making,” April 2023.

[3] Diwas Sharma and Udit Kr. Chakraborty, “An Improved Chess Machine based on Artificial Neural Networks” December 2014

[4] Reid McIlroy-Young, Siddhartha Sen, Jon Kleinberg, Ashton Anderson, “Aligning Superhuman AI with Human Behaviour: Chess as a Model System”, July 2020.

[5] Rohit Verma and Sanchit Verma, “AI Based Chess”, May 2022.

[7] Magesh Surya Ambikapathi, “Modelling Chess AI using Deep Learning – Survey” July 2022.

[8] Xiaoyu Zhang, Wei Liu, Linhui Lou and Fangchun Yang, “AI Enabled Bridge Bidding Supporting Interactive Visualization”, February 2022.

[9] Francisco S. Marcondes, Dalila Durães, Flávio Santos, José João Almeida andPauloNovais, “Neural Network Explainable AI Based on Paraconsistent Analysis: An Extension” October 2021.

[10] Azlan Iqbal, “A New Chess Variant for Gaming AI”, 2015