

INTEGRAÇÃO DE IA COM CLP EM UM BRAÇO MECÂNICO COM RECONHECIMENTO DE IMAGEM

INTEGRATION OF AI WITH PLC IN A MECHANICAL ARM WITH IMAGE RECOGNITION

Bruno Martins Mariano de Andrade Santos¹

Victor Hugo D. P. Jacinto Pedroso²

Felipe Ferreira Faustino³

Jean Santos Vinícius Ribeiro⁴

Resumo: Objetivo: desenvolver e verificar uma arquitetura de integração entre inteligência artificial, visão computacional, controlador lógico programável e manipulador robótico capaz de reconhecer o estado de um tabuleiro de xadrez, selecionar uma jogada e coordenar sua execução física. Método: foi realizado um estudo aplicado de desenvolvimento tecnológico, em 2026, organizado em cinco camadas funcionais: percepção por sensor de visão industrial, integração por programa em Python, decisão pelo motor Stockfish, coordenação e segurança por controlador Siemens S7-1200 e atuação por controlador embarcado e manipulador SCARA. A verificação concentrou-se na lógica de software, nos protocolos de comunicação, no tratamento de falhas e na simulação integrada. Resultados: os 72 testes automatizados do programa de integração foram aprovados, assim como os conjuntos de 25 e 61 testes do programa da célula e do software de bancada; o núcleo de movimento também foi verificado por 37 testes. A limitação de tempo da comunicação reduziu uma espera bloqueada de 5,0 s para menos de 1,5 s. Na simulação integrada, o travamento do motor foi tratado em 4,3 s e o travamento do laço principal foi detectado em 4,0 s, com preservação do estado da partida

¹ Engenheiro eletricitista, Universidade Mogi Das Cruzes ORCID: 0009-0004-6852-9207

² Engenheiro eletricitista, Universidade Mogi Das Cruzes ORCID: 0009-0007-8668-0009

³ Engenheiro eletricitista, Universidade Mogi Das Cruzes ORCID: 0009-0006-2819-8700

⁴ Engenheiro eletricitista, universidade Mogi Das Cruzes

após o reinício. Conclusão: os resultados sustentam a consistência funcional da arquitetura e sua capacidade de recuperação em software e simulação, mantendo o controlador industrial como elemento de coordenação e segurança. A avaliação física de visão, posicionamento, tempo de resposta e partidas completas ainda é necessária para caracterizar o desempenho global da célula.

Palavras-chave: Automação industrial; Inteligência artificial; Visão computacional; Controlador lógico programável; Robótica

Abstract: Objective: to develop and verify an integration architecture combining artificial intelligence, computer vision, a programmable logic controller, and a robotic manipulator to recognize a chessboard state, select a move, and coordinate its physical execution. Method: an applied technological development study was conducted in 2026 and organized into five functional layers: perception by an industrial vision sensor, integration through a Python application, decision-making by the Stockfish engine, coordination and safety by a Siemens S7-1200 programmable logic controller, and actuation by an embedded controller and a SCARA manipulator. Verification focused on software logic, communication protocols, fault handling, and integrated simulation. Results: all 72 automated tests of the integration application passed, as did the 25- and 61-test sets for the cell program and bench software; the motion core was also verified by 37 tests. Communication timeout control reduced a blocked wait from 5.0 s to less than 1.5 s. In integrated simulation, engine lockup was handled in 4.3 s and a stalled main loop was detected in 4.0 s, while the game state was preserved after restart. Conclusion: the results support the functional consistency and recovery capability of the architecture in software and simulation while keeping the industrial controller responsible for coordination and safety. Physical evaluation of vision, positioning, response time, and complete games is still required to characterize overall cell performance.

Keywords: Industrial automation; Artificial intelligence; Computer vision; Programmable logic controller; Robotics.

INTRODUÇÃO

A automação industrial vem incorporando, de forma crescente, recursos capazes de perceber o ambiente, interpretar informações e apoiar decisões. Com a Indústria 4.0, tecnologias de inteligência artificial, sistemas ciberfísicos e conectividade passaram a se aproximar do nível de controle, formando arquiteturas nas quais informação e ação física precisam permanecer sincronizadas (Dafflon; Moalla; Ouzrout, 2021; Gao et al., 2024; Oks et al., 2024; Pivoto et al., 2021; Rakholia et al., 2024). Nesse cenário, o controlador lógico programável continua tendo papel central por sua robustez, previsibilidade de varredura e integração com dispositivos de campo. Ao mesmo tempo, tarefas de maior custo computacional podem ser executadas em plataformas externas, desde que a comunicação seja bem definida e existam mecanismos de supervisão, tratamento de falhas e recuperação (Majumder; Shakil; Zoitl, 2022; Mellado; Núñez, 2022).

Nesse contexto, a visão computacional funciona como uma ponte entre o ambiente físico e o sistema de controle, transformando imagens em informações utilizáveis pela automação. Em aplicações industriais, porém, a qualidade da leitura não depende apenas do algoritmo: iluminação, contraste, enquadramento, calibração e estabilidade geométrica também influenciam diretamente o resultado (Bonci et al., 2021; De Ketelaere et al., 2022; Gao et al., 2022; Ren et al., 2022). Quando a percepção visual é associada à robótica, surge uma questão adicional de segurança: uma leitura incorreta pode ser convertida em movimento físico. Por essa razão, a arquitetura precisa prever inconsistências de estado, limites de tempo, intertravamentos e procedimentos de recuperação (Dzedzickis et al., 2022; Javaid et al., 2021).

O xadrez oferece uma bancada adequada para observar essa integração. O tabuleiro possui geometria conhecida, regras formais, estados discretos e movimentos que podem ser conferidos após cada ação. Trabalhos recentes já demonstraram sistemas autônomos que reúnem visão computacional, motores de xadrez e braços robóticos, confirmando a viabilidade de combinar percepção, decisão e manipulação física (Alkhatib et al., 2026; Phuc;

Son, 2025). Em muitas dessas soluções, entretanto, a coordenação permanece concentrada no computador. Para uma aplicação de automação, interessa explorar outra divisão de responsabilidades: deixar o processamento inteligente no computador e manter no controlador industrial o sequenciamento, os limites de tempo, os intertravamentos e a autorização dos movimentos.

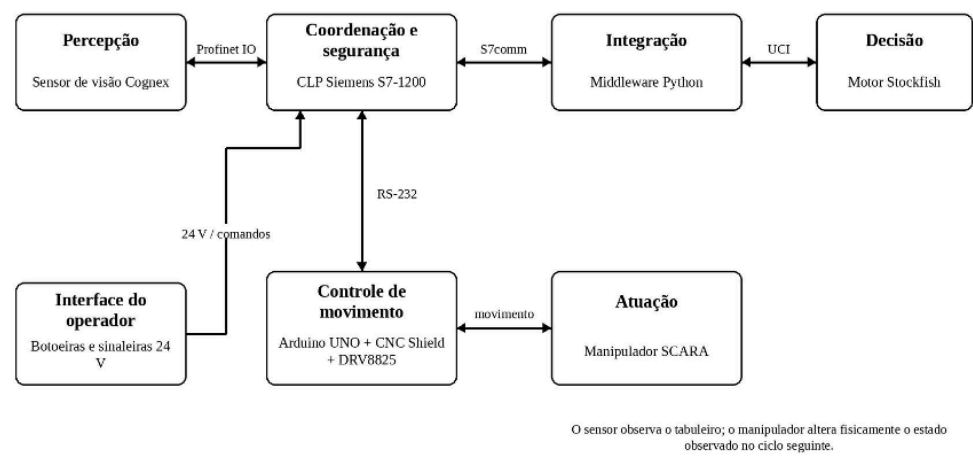
O tabuleiro foi adotado como bancada de uma arquitetura modular com visão industrial, integração em Python, motor Stockfish, CLP Siemens S7-1200, Arduino e manipulador SCARA. O estudo examinou a troca de informações, a preservação do estado da partida e a resposta a leituras inválidas, falhas de comunicação e travamentos. O objetivo foi desenvolver e verificar essa arquitetura para reconhecer o tabuleiro, decidir jogadas e coordenar sua execução física.

MATERIAL E MÉTODOS

Trata-se de estudo aplicado de desenvolvimento tecnológico, realizado em 2026 em laboratório universitário de automação no estado de São Paulo. Sem participação de seres humanos ou coleta de dados pessoais, a unidade de análise foi o próprio sistema. A verificação considerou a consistência dos dados, lances válidos e inválidos, sinal de vida, tempos de indisponibilidade, preservação do estado da partida e respostas da máquina de estados. A análise foi descritiva, com aprovação ou reprovação dos testes e comparação dos tempos medidos com os limites de projeto. Os ensaios físicos de visão, posicionamento, tempo de resposta e partidas completas ficaram para etapa posterior, sem inferência de resultados.

A arquitetura foi dividida em percepção, com sensor Cognex; integração e representação da partida em Python; decisão pelo Stockfish; coordenação e segurança no Siemens S7-1200; e atuação por Arduino UNO, CNC Shield V3, drivers DRV8825 e manipulador SCARA. Essa separação mantém as tarefas não determinísticas fora do controlador industrial sem retirar dele a autoridade sobre a sequência e os intertravamentos. A Figura 1 apresenta a arquitetura e seus principais enlaces de comunicação.

Figura 1 – Arquitetura funcional da célula robótica e principais enlaces de comunicação



Fonte: elaboração própria.

A Tabela 1 resume os equipamentos que participam diretamente do ciclo de operação. O projeto foi estruturado com interfaces explícitas entre as camadas, permitindo testar cada subsistema de forma isolada ou substituí-lo sem alterar a lógica global, desde que o contrato de dados seja mantido. Essa organização é compatível com arquiteturas ciberfísicas e com o uso de camadas intermediárias para integrar controladores industriais a aplicações de nível superior (Dafflon; Moalla; Ouzrout, 2021; Majumder; Shakil; Zoitl, 2022; Mellado; Núñez, 2022; Oks et al., 2024; Pivoto et al., 2021; Rakholia et al., 2024).

Tabela 1 – Subsistemas, equipamentos e funções principais

Camada	Equipamento	Função principal
Percepção	Sensor de visão Cognex	Aquisição da imagem do tabuleiro e classificação das 64 casas quanto à ocupação e à cor observada.
Integração	Computador com <i>middleware</i> Python	Manutenção do estado da partida, identificação do lance humano, comunicação com o motor e decomposição do lance do robô.

Decisão	Motor Stockfish	Cálculo do lance de resposta a partir da posição corrente.
Coordenação e segurança	CLP Siemens S7-1200 com módulo CM 1241 RS-232	Execução da máquina de estados, disparo da visão, supervisão de tempos, troca de dados e autorização dos movimentos.
Atuação	Arduino UNO, CNC Shield V3, DRV8825 e manipulador SCARA	Referenciamento dos eixos, execução das trajetórias e manipulação física das peças.
Interface do operador	Botoeiras e sinaleiras em 24 V	Confirmação do lance, partida, rearme e indicação dos estados da célula.

Fonte: elaboração própria.

O sistema de visão foi concebido para observar o tabuleiro por meio de uma câmera instalada de forma rígida e com iluminação controlada. Foram definidas 64 regiões de interesse, uma para cada casa. Em cada região, o sistema produz dois atributos: presença de peça e classe de cor. Em vez de tentar reconhecer visualmente o tipo de cada peça, o programa mantém internamente a posição completa da partida. Depois que o operador realiza um movimento, o estado observado é comparado aos estados que resultariam de todos os lances legais disponíveis na posição corrente. O lance humano é aceito quando há correspondência com um desses estados. Dessa forma, reduz-se a complexidade da tarefa visual e transfere-se a validação das regras para a biblioteca de xadrez usada pelo programa. A estratégia é compatível com a percepção por regiões e com a necessidade de controlar variações de iluminação e posicionamento (Bonci et al., 2021; De Ketelaere et al., 2022; Gao et al., 2022; Szeliski, 2022).

Para evitar que uma leitura transitória seja aceita como estado válido, o controlador solicita duas aquisições consecutivas e só encaminha a observação quando os resultados coincidem. O bit de cor de uma casa vazia é ignorado, pois não possui significado físico. Lances simples, capturas, roque e captura en passant são diferenciados pelas mudanças de ocupação no tabuleiro. No protótipo, as promoções são tratadas como promoção para dama.

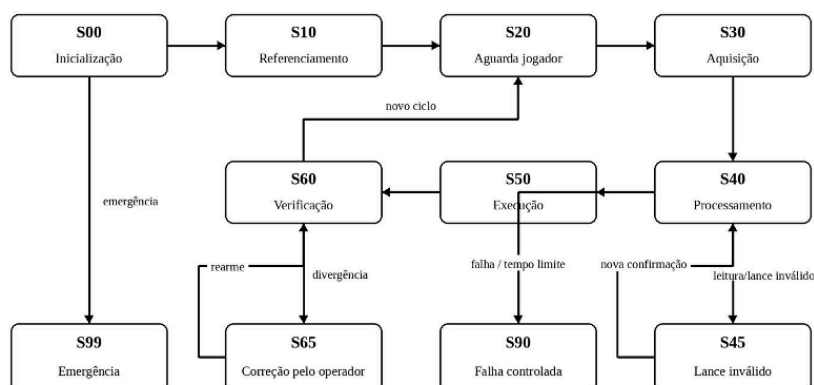
Qualquer vetor de anomalia ou estado incompatível com um lance legal devolve uma

condição de correção ao operador. Depois de validar o lance humano, o programa consulta o Stockfish, atualiza o estado esperado e decompõe a resposta em transferências elementares entre origem e destino. A representação formal dos estados e a busca do motor seguem princípios consolidados de inteligência artificial aplicada a jogos e de percepção visual (Russell; Norvig, 2021; Szeliski, 2022).

As comunicações foram distribuídas de acordo com a natureza de cada enlace. O sensor de visão e o controlador industrial trocam dados por Profinet IO; o controlador e o programa em Python utilizam S7comm, por meio da biblioteca python-snap7; o programa e o motor de xadrez se comunicam pelo protocolo UCI; e o controlador industrial se conecta ao controlador de movimento por RS-232 em modo livre, com transceptor de nível apropriado. Botoeiras, sinaleiras e cadeia de emergência operam em 24 V. Entre o controlador e o programa, a troca de dados ocorre por um bloco dedicado e utiliza números de sequência para requisição e confirmação. Um resultado só é aceito quando a confirmação corresponde à requisição ativa, evitando que dados parcialmente escritos sejam interpretados como válidos. Esse mecanismo se aproxima das práticas de integração por middleware utilizadas em ambientes industriais heterogêneos (Majumder; Shakil; Zoitl, 2022; Mellado; Núñez, 2022; Seong et al., 2023).

O sequenciamento foi implementado no controlador por meio de uma máquina de estados. Na inicialização, o sistema verifica o sinal de vida do programa, a comunicação com a visão e a ausência de emergência. Em seguida, executa referenciamento, espera do jogador, aquisição, processamento, movimento e verificação do tabuleiro. Leituras inválidas levam a uma nova confirmação; divergências após o movimento do robô direcionam o processo para correção pelo operador; perdas de comunicação ou estouros de tempo levam a uma condição controlada de falha; e a parada de emergência possui prioridade global. A Figura 2 resume o fluxo principal. A lógica de segurança elétrica foi tratada de acordo com princípios de parada e isolamento aplicáveis a equipamentos de máquinas (International Electrotechnical Commission, 2016).

Figura 2 – Síntese da máquina de estados e dos caminhos de recuperação



A parada de emergência tem prioridade sobre a sequência. Falhas que comprometem a consistência do estado físico e lógico levam à condição controlada de falha.

Fonte: elaboração própria.

O controlador de movimento foi programado em C++ para o ATmega328P. Os quatro eixos recebem pulsos gerados por interrupção periódica de 20 kHz e são sincronizados por uma implementação do algoritmo de Bresenham, com rampas de aceleração e desaceleração. As posições calibradas das casas, da posição de repouso e do ponto de descarte são armazenadas em memória não volátil e protegidas por verificação CRC32. Antes de cada partida, o manipulador realiza o referenciamento pelos fins de curso. Durante o ciclo, o controlador industrial envia apenas os índices de origem e destino, enquanto o movimento de baixo nível permanece encapsulado no controlador embarcado. Essa separação entre planejamento, coordenação e acionamento é compatível com arquiteturas robóticas em que diferentes níveis assumem responsabilidades distintas (Corke, 2017; Dzedzickis et al., 2022; Javaid et al., 2021).

A verificação documentada concentrou-se em testes automatizados e simulação. O programa de integração reuniu 72 testes de bloco de dados, lógica do jogo, serviço e protocolo, motor de decisão, sinal de vida, comunicação S7 e célula simulada. O Stockfish real e um motor UCI simulado permitiram reproduzir travamentos; a comunicação foi testada

com servidor S7 simulado e intermediário de rede para representar ausência de resposta. O programa da célula teve 25 testes, o programa de bancada, 61, e o núcleo de movimento e o protocolo embarcado, 37 testes executados no computador. Ensaios físicos de visão, posicionamento, tempo de resposta, centralização das peças e partidas completas estavam previstos, mas não tinham resultados finais registrados; portanto, nenhum desses dados foi inferido.

RESULTADOS E DISCUSSÃO

Os resultados mostram, em primeiro lugar, que a arquitetura modular pode ser exercitada sem depender de todos os elementos físicos ao mesmo tempo. O ciclo lógico foi executado com câmera, robô e operador simulados sem alterar o código responsável pelas regras do jogo e pela orquestração das requisições. Na prática, isso permite substituir temporariamente percepção, decisão ou movimento por simuladores durante o desenvolvimento, enquanto o controlador industrial mantém a mesma lógica de estados. A solução também impede que o motor de xadrez, cujo tempo de busca varia conforme o nível escolhido, comande diretamente as saídas físicas. Essa organização aproxima o protótipo de uma arquitetura de automação em que aplicações inteligentes complementam, e não substituem, o controlador responsável pelo processo (Dafflon; Moalla; Ouzrout, 2021; Gao et al., 2024; Majumder; Shakil; Zoitl, 2022; Mellado; Núñez, 2022; Oks et al., 2024; Pivoto et al., 2021; Rakholia et al., 2024).

Os 72 testes automatizados do programa de integração foram aprovados. Como mostra a Tabela 2, a suíte abrange desde a codificação do bloco de dados até a célula simulada. Os conjuntos adicionais do programa da célula e do software de bancada, com 25 e 61 casos, respectivamente, também foram aprovados. Esse resultado é importante porque as falhas relevantes do protótipo não se limitam ao cálculo da jogada: envolvem ordem de bytes, duplicidade de requisição, reinício do controlador, retenção de comando, persistência do estado e reação a perdas de comunicação. Verificar esses cenários antes do comissionamento físico reduz o risco de uma inconsistência de software resultar em movimento indevido do

manipulador.

Tabela 2 – Suíte de testes automatizados do programa de integração

Grupo	Testes	Aspectos verificados
Bloco de dados	7	Deslocamentos, ordem de bytes, vetores de bits, limites e bloqueio de escrita em regiões indevidas.
Lógica de jogo	20	Lance simples, captura, roque, <i>en passant</i> , promoção, lance ilegal, xeque, anomalia, decomposição e persistência.
Serviço e protocolo	25	Confirmação por sequência, duplicidade, republicação, retenção de requisição, reinícios e falhas controladas.
Motor de decisão	11	Travamento, recuperação, inicialização lenta, prazo insuficiente e validação de configuração.
Sinal de vida	3	Continuidade durante consulta prolongada ao motor e interrupção com laço principal travado.
Comunicação S7	2	Leitura, escrita e tempo máximo diante de controlador sem resposta.
Célula simulada	4	Ciclo completo, queda de peça, lance inválido e emergência.

Fonte: elaboração própria.

Durante os testes, foi identificado um acoplamento inadequado entre o sinal de vida e o laço principal. Com uma consulta de 3 s ao motor, atualizar o contador no próprio laço produziria uma interrupção aparente maior que o limite de 2 s supervisionado pelo controlador. No ensaio automatizado, a rotina paralela desativada resultou em intervalo de 2,99 s e o teste foi reprovado. Depois que a atualização do sinal de vida foi separada do laço principal, o intervalo passou a atender ao critério de menos de 0,5 s. A mesma preocupação

foi aplicada à comunicação S7: uma leitura que permanecia bloqueada por 5,0 s diante de um controlador simulado sem resposta passou a falhar em menos de 1,5 s após a definição de limite de 1 s para a operação. Os resultados evidenciam que temporização e isolamento de tarefas são tão relevantes quanto a lógica funcional quando aplicações de nível superior são integradas a controladores industriais (Dafflon; Moalla; Ouzrout, 2021; Majumder; Shakil; Zoitl, 2022; Mellado; Núñez, 2022).

A simulação integrada também permitiu acompanhar a recuperação diante de falhas capazes de comprometer a disponibilidade ou a consistência da célula. A Tabela 3 reúne os casos mais representativos. O travamento do motor em duas tentativas levou a uma condição controlada de falha, com código de diagnóstico, em 4,3 s; o travamento do laço principal foi detectado em 4,0 s; a requisição permaneceu retida quando o controlador foi colocado em STOP; e o reinício do programa durante a partida preservou o estado do jogo. Quando uma peça caiu, o sistema desviou para a etapa de correção e exigiu novo referenciamento antes da continuidade. Esses comportamentos reforçam que, em um sistema ciberfísico, detectar uma anomalia não é suficiente: também é necessário conduzir o processo a um estado seguro e permitir uma retomada previsível (Dafflon; Moalla; Ouzrout, 2021; Oks et al., 2024; Pivoto et al., 2021; Rakholia et al., 2024).

Tabela 3 – Resultados observados na injeção de falhas em simulação

Cenário	Resultado observado	Interpretação
Queda de peça	Transição para correção pelo operador e novo referenciamento após o rearme.	A inconsistência entre estado lógico e físico não é aceita como continuação normal.
Travamento do motor de decisão	Falha controlada após duas tentativas, com código de diagnóstico, em 4,3 s.	O orçamento de tempo evita bloqueio indefinido da sequência.
Controlador em STOP com requisição pendente	Requisição mantida até o retorno do sinal de vida.	Evita perda ou reprocessamento indevido da transação.

Travamento do laço principal	Detecção pelo controlador em 4,0 s.	O sinal de vida independente permite distinguir cálculo demorado de travamento real.
Reinício do programa de integração	Estado da partida preservado e retomado.	A persistência reduz a necessidade de reiniciar todo o processo.

Fonte: elaboração própria.

Em relação aos robôs de xadrez descritos recentemente, a contribuição deste trabalho está menos no desempenho enxadrístico e mais na distribuição das responsabilidades de controle. Estudos de 2025 e 2026 já integraram visão computacional, Stockfish e manipuladores, inclusive um SCARA de quatro graus de liberdade, demonstrando a viabilidade de partidas autônomas (Alkhatib et al., 2026; Phuc; Son, 2025). Neste estudo, o computador interpreta o tabuleiro e calcula a resposta, enquanto o CLP mantém o sequenciamento, a supervisão de tempos, os intertravamentos e a autorização dos movimentos. Essa divisão aproxima a inteligência artificial da automação industrial sem transferir toda a autoridade de controle para um sistema operacional de propósito geral.

Inferir o lance a partir de ocupação e cor, em vez de reconhecer visualmente cada tipo de peça, também foi uma escolha de engenharia. A estratégia reduz a complexidade da visão, mas depende de condições conhecidas: a partida deve começar na posição inicial prevista, todos os lances precisam passar pelo programa, peças capturadas devem ser retiradas corretamente e o tabuleiro precisa permanecer estável do ponto de vista geométrico e de iluminação. Ao limitar a percepção ao necessário para distinguir transições legais, parte da inteligência é transferida para o modelo interno da partida. A abordagem atende ao protótipo proposto, mas não substitui o reconhecimento completo quando é necessário reconstruir uma posição arbitrária ou recuperar-se de alterações que não tenham sido registradas (Alkhatib et al., 2026; Bonci et al., 2021; De Ketelaere et al., 2022; Gao et al., 2022; Phuc; Son, 2025; Szeliski, 2022).

Os resultados sustentam a lógica, os protocolos e a recuperação, mas não o desempenho físico global. Os ensaios previstos avaliam visão, posicionamento, tempo de

resposta, quedas de peças, garra, centralização magnética e partidas completas. Como esses dados não estavam preenchidos, não se apresentam precisão, médias, dispersão ou taxa de sucesso. A conclusão restringe-se à simulação; a validação física depende desses ensaios.

CONCLUSÃO

O objetivo do estudo foi alcançado no que diz respeito ao desenvolvimento da arquitetura e à verificação de sua lógica de integração. A solução reuniu percepção visual, representação da partida, motor de decisão, controlador lógico programável e controlador de movimento em uma cadeia modular, mantendo no controlador industrial a coordenação do processo e os mecanismos de segurança. A comparação entre o estado observado e os estados produzidos por lances legais simplificou a identificação do movimento humano, enquanto a máquina de estados permitiu tratar de forma explícita lances inválidos, divergências no tabuleiro, perdas de comunicação, travamentos do motor e situações de emergência. Os testes automatizados e a simulação integrada confirmaram a consistência dos protocolos, a persistência do estado e as respostas previstas para as falhas avaliadas. Esse arranjo também manteve a autoridade do CLP separada do tempo variável de processamento do motor de xadrez, reforçando a supervisão do ciclo mesmo quando a decisão computacional exige maior tempo de execução. Ainda assim, os dados disponíveis não permitem estender essa conclusão ao desempenho físico completo do manipulador e do sistema de visão. Para uma validação experimental integral, permanecem necessários os ensaios físicos previstos no TCC, especialmente classificação das casas, precisão de posicionamento, tempo de resposta, falhas de pega e partidas completas, com registro do número de repetições, medidas de tendência e dispersão e comparação com os critérios de aceitação.

AGRADECIMENTOS E FINANCIAMENTO

Os autores agradecem a Deus pelo apoio ao longo do desenvolvimento deste trabalho.

A pesquisa não recebeu financiamento externo.

DECLARAÇÃO SOBRE O USO DE INTELIGÊNCIA ARTIFICIAL GENERATIVA

Durante a preparação deste manuscrito, os autores utilizaram o ChatGPT, da OpenAI, como apoio à organização do texto, à síntese, à revisão linguística e à adequação editorial. Todo o conteúdo produzido com esse apoio foi revisado e editado pelos autores, que assumem integral responsabilidade pela versão submetida. A ferramenta não foi utilizada para gerar, modificar ou analisar dados experimentais.

CONFLITO DE INTERESSES

Os autores declaram não possuir conflitos de interesse relacionados a este estudo.

DISPONIBILIDADE DOS DADOS

Os registros gerados pelo sistema, incluindo arquivos de eventos em formato CSV e registros de partidas em formato PGN, permanecem sob guarda dos autores e poderão ser disponibilizados mediante solicitação razoável. O documento de origem não informa depósito desses dados em repositório público com identificador persistente.

REFERÊNCIAS

GAO, R. X. et al. Artificial intelligence in manufacturing: state of the art, perspectives, and future directions. *CIRP Annals*, v. 73, n. 2, p. 723–749, 2024. DOI:

10.1016/j.cirp.2024.04.101. Disponível em: <https://doi.org/10.1016/j.cirp.2024.04.101>. Acesso em: 6 out. 2026.

OKS, S. J. et al. Cyber-physical systems in the context of Industry 4.0: a review, categorization and outlook. *Information Systems Frontiers*, v. 26, p. 1731–1772, 2024. DOI:

10.1007/s10796-022-10252-x. Disponível em: <https://doi.org/10.1007/s10796-022-10252-x>.

Acesso em: 6 out. 2026.

PIVOTO, D. G. S. et al. Cyber-physical systems architectures for industrial internet of things applications in Industry 4.0: a literature review. *Journal of Manufacturing Systems*, v.

58, p. 176–192, 2021. DOI: 10.1016/j.jmsy.2020.11.017. Disponível em: <https://doi.org/10.1016/j.jmsy.2020.11.017>. Acesso em: 6 out. 2026.

RAKHOLIA, R. et al. Advancing manufacturing through artificial intelligence: current landscape, perspectives, best practices, challenges, and future direction. *IEEE Access*, v. 12, p.

131621–131637, 2024. DOI: 10.1109/ACCESS.2024.3458830. Disponível em: <https://doi.org/10.1109/ACCESS.2024.3458830>. Acesso em: 6 out. 2026.

DAFFLON, B.; MOALLA, N.; OUZROUT, Y. The challenges, approaches, and used techniques of CPS for manufacturing in Industry 4.0: a literature review. *The International Journal of Advanced Manufacturing Technology*, v. 113, p. 2395–2412, 2021. DOI:

10.1007/s00170-020-06572-4. Disponível em: <https://doi.org/10.1007/s00170-020-06572-4>. Acesso em: 6 out. 2026.

MELLADO, J.; NÚÑEZ, F. Design of an IoT-PLC: a containerized programmable logical controller for the Industry 4.0. *Journal of Industrial Information Integration*, v. 25, art.

100250, 2022. DOI: 10.1016/j.jii.2021.100250. Disponível em: <https://doi.org/10.1016/j.jii.2021.100250>. Acesso em: 6 out. 2026.

MAJUMDER, M.; SHAKIL, M.; ZOITL, A. Evaluation of PLC4X based middleware as integrator of brownfield systems into industrial cyber-physical systems. *IFAC-PapersOnLine*, v. 55, n. 2, p. 427–432, 2022. DOI: 10.1016/j.ifacol.2022.04.231. Disponível em: <https://doi.org/10.1016/j.ifacol.2022.04.231>. Acesso em: 6 out. 2026.

BONCI, A. et al. Human-robot perception in industrial environments: a survey. *Sensors*, v. 21, n. 5, art. 1571, 2021. DOI: 10.3390/s21051571. Disponível em: <https://doi.org/10.3390/s21051571>. Acesso em: 6 out. 2026.

DE KETELAERE, B. et al. A fresh look at computer vision for industrial quality control.

Quality Engineering, v. 34, n. 1, p. 152–158, 2022. DOI: 10.1080/08982112.2021.2001828. Disponível em: <https://doi.org/10.1080/08982112.2021.2001828>. Acesso em: 6 out. 2026.

GAO, Y. et al. A review on recent advances in vision-based defect recognition towards industrial intelligence. *Journal of Manufacturing Systems*, v. 62, p. 753–766, 2022. DOI:

10.1016/j.jmsy.2021.05.008. Disponível em: <https://doi.org/10.1016/j.jmsy.2021.05.008>. Acesso em: 6 out. 2026.

DZEDZICKIS, A. et al. Advanced applications of industrial robotics: new trends and possibilities. *Applied Sciences*, v. 12, n. 1, art. 135, 2022. DOI: 10.3390/app12010135. Disponível em: <https://doi.org/10.3390/app12010135>. Acesso em: 6 out. 2026.

JAVAID, M. et al. Substantial capabilities of robotics in enhancing Industry 4.0 implementation. *Cognitive Robotics*, v. 1, p. 58–75, 2021. DOI: 10.1016/j.cogr.2021.06.001. Disponível em: <https://doi.org/10.1016/j.cogr.2021.06.001>. Acesso em: 6 out. 2026.

ALKHATIB, M. et al. Automated chess gameplay with computer vision and a robotic arm. *Engineering Reports*, v. 8, n. 2, e70607, 2026. DOI: 10.1002/eng2.70607. Disponível em: <https://doi.org/10.1002/eng2.70607>. Acesso em: 6 out. 2026.

PHUC, T. D.; SON, B. C. Development of an autonomous chess robot system using computer vision and deep learning. *Results in Engineering*, v. 25, art. 104091, 2025. DOI:

10.1016/j.rineng.2025.104091. Disponível em: <https://doi.org/10.1016/j.rineng.2025.104091>. Acesso em: 6 out. 2026.

SZELISKI, R. *Computer vision: algorithms and applications*. 2. ed. Cham: Springer, 2022.

RUSSELL, S. J.; NORVIG, P. *Artificial intelligence: a modern approach*. 4. ed. Harlow: Pearson, 2021.

CORKE, P. *Robotics, vision and control: fundamental algorithms in MATLAB*. 2. ed. Cham: Springer, 2017.

SEONG, J. et al. Enhancing industrial communication with Ethernet/Internet Protocol: a study and analysis of real-time cooperative robot communication and automation via Transmission Control Protocol/Internet Protocol. *Sensors*, v. 23, n. 20, art. 8580, 2023. DOI:

10.3390/s23208580. Disponível em: <https://doi.org/10.3390/s23208580>. Acesso em: 6 out. 2026.

INTERNATIONAL ELECTROTECHNICAL COMMISSION. IEC 60204-1:2016: safety of machinery - electrical equipment of machines - Part 1: general requirements. Geneva: IEC, 2016.

REN, Z. et al. State of the art in defect detection based on machine vision. *International Journal of Precision Engineering and Manufacturing-Green Technology*, v. 9, n. 2, p. 661–691, 2022. DOI: 10.1007/s40684-021-00343-6. Disponível em: <https://doi.org/10.1007/s40684-021-00343-6>. Acesso em: 6 out. 2026.