

# Minichess com Machine Learning e Sistema CNC

**Fernando Frare Vieira**

[fernandofrarevieira@alunos.utfpr.edu.br](mailto:fernandofrarevieira@alunos.utfpr.edu.br)

+55 41 9900-6940

**Gabriel Affonso Borges Caballero**

[gabriellaffonso@alunos.utfpr.edu.br](mailto:gabriellaffonso@alunos.utfpr.edu.br)

+55 41 9151-3335

**Marco Vieira Buseti**

[marcovieirabuseti@alunos.utfpr.edu.br](mailto:marcovieirabuseti@alunos.utfpr.edu.br)

+55 41 99798-4695

12 de abril de 2025

## 1 Introdução

O projeto "Minichess com Machine Learning e Sistema CNC" foi desenvolvido com o propósito pedagógico de introduzir conceitos de Machine Learning para crianças. Utilizando um tabuleiro de xadrez simplificado (4x4), o sistema integra componentes de Inteligência Artificial e automação para demonstrar, de forma lúdica, como o aprendizado de máquina pode ser aplicado na prática. A solução proposta conta com um módulo de Visão Computacional, que realiza a captura do tabuleiro físico, e com um sistema CNC, que executa a movimentação automatizada das peças através da comunicação entre um computador e um Arduino.

## 2 Motivação

O principal objetivo deste projeto é educar crianças sobre os fundamentos do Machine Learning, uma área de extrema relevância na atualidade, por meio de uma experiência interativa e prática. Além disso, o Minichess 4x4 visa estimular o desenvolvimento do raciocínio lógico e da tomada de decisão, utilizando o xadrez simplificado como ferramenta de aprendizado. A integração dos componentes físicos (tabuleiro e sistema CNC) com a Inteligência Artificial oferece um ambiente dinâmico, onde o aluno interage diretamente com o jogo, consolidando os conceitos através da prática.

### 3 Descrição Geral

O Minichess 4x4 foi concebido para funcionar de forma integrada, utilizando um computador, uma câmera e um sistema CNC. Diferente de soluções que operam apenas por meio de interfaces gráficas, o jogo é realizado de modo físico: a criança posiciona as peças no tabuleiro real e o sistema, por meio da captura de imagem, interpreta a configuração atual e, em conjunto com a Inteligência Artificial, toma a decisão sobre o movimento. Posteriormente, o sistema CNC, controlado via comunicação serial com o Arduino, efetua a movimentação física das peças. Essa abordagem garante uma experiência completa e interativa, unindo teoria e prática.

A integração é realizada através de APIs desenvolvidas em Python, utilizando o framework Flask para comunicação entre os módulos de Visão Computacional, Inteligência Artificial e controle CNC. O sistema CNC é baseado em um manipulador cartesiano com três graus de liberdade (eixos X, Y e Z), utilizando motores de passo Nema 17 em configuração CoreXY para movimentação precisa no plano horizontal, e um servo motor SG90 para controle vertical do eletroímã responsável pela movimentação das peças.

## 4 Diagrama de Blocos

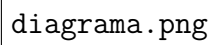


diagrama.png

Figura 1: Diagrama de Blocos do Sistema Minichess

## 5 Descrição do Software

O software deste projeto é responsável por:

- Implementar a lógica do jogo e a Inteligência Artificial com Machine Learning, utilizando o algoritmo **Q-learning** (aprendizado por reforço) para tomada de decisão. O agente aprende a otimizar suas jogadas através de um processo iterativo, onde cada ação é associada a um valor (\*Q-value\*) que representa a recompensa esperada em longo prazo. O algoritmo equilibra exploração (testar novas jogadas) e exploração (usar conhecimento adquirido), ajustando-se dinamicamente conforme interage com o jogador. A função de recompensa é definida para priorizar vitórias, evitar derrotas e incentivar movimentos estratégicos no tabuleiro 4x4.
- Capturar e processar imagens do tabuleiro físico utilizando a biblioteca OpenCV. As técnicas incluem detecção de cores (espaço HSV), filtragem de bordas (Canny) e comparação de contornos para identificar a posição das peças. A imagem é recortada para isolar o tabuleiro e evitar interferências externas.
- Gerenciar a comunicação entre o computador e o Arduino via protocolo serial, enviando comandos em G-Code para controle do sistema CNC. Uma API REST (desenvolvida com Flask) coordena o fluxo entre os subsistemas: após a jogada do usuário, a visão computacional envia o estado do tabuleiro para a IA, que retorna a ação aprendida pelo Q-learning, convertida em coordenadas para movimentação física pelo CNC.

## 6 Descrição do Hardware

Para a realização do jogo físico, o sistema conta com os seguintes componentes de hardware:

- **Computador com Câmera:** Utilizado para capturar imagens do tabuleiro (resolução 1920x1080) e processar as informações necessárias para a tomada de decisão. A câmera é fixada em uma estrutura de suporte 3D impressa, posicionada verticalmente para garantir visibilidade total do tabuleiro.
- **Arduino UNO com CNC Shield V3:** Responsável por controlar os motores de passo (Nema 17) e o servo motor SG90. A CNC Shield utiliza drivers A4988 configurados em modo 1/4 de passo para equilibrar velocidade e precisão, e um relé JQC-3FF-S-Z para acionamento do eletroímã de 12V.
- **Sistema CNC CoreXY:** Composto por dois motores de passo para movimento no plano XY, seguindo o arranjo CoreXY para reduzir a inércia. O eixo Z é controlado por um servo motor, responsável por abaixar e levantar o eletroímã. A estrutura mecânica é baseada em perfis de MDF cortados a laser, com correias sincronizadas para transmissão de movimento.
- **Fonte de Alimentação:** Duas fontes de 12V 5A são utilizadas: uma para os motores de passo e ventoinhas de refrigeração, e outra exclusiva para o eletroímã.

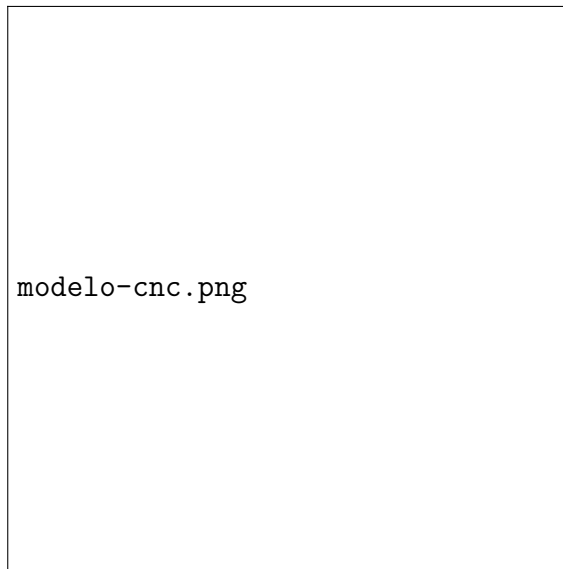


Figura 2: Referência do sistema CNC para movimentação física

## 7 Lista de Componentes

- **Software:**

- Python 3.8+ (com bibliotecas OpenCV, Flask, NumPy, e PySerial)
- Firmware GRBL (para interpretação de G-Code no Arduino)

- **Hardware:**

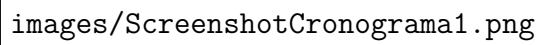
- Computador com câmera USB Full HD
- Arduino UNO + CNC Shield V3 + Drivers A4988
- 2x Motores de Passo Nema 17 + Correias GT2
- Servo Motor SG90 + Eletroímã 12V
- Fonte de Alimentação 12V 5A (2 unidades)
- Estrutura em MDF cortada a laser + Suportes 3D impressos

## 8 Integração entre Software e Hardware

A comunicação entre os módulos é realizada através de APIs desenvolvidas em Flask, garantindo modularidade e escalabilidade. O fluxo segue três etapas:

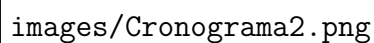
1. **Visão Computacional:** Após a jogada do usuário, a câmera captura uma imagem, processada pelo OpenCV para gerar uma matriz 4x4 representando o tabuleiro.
2. **Inteligência Artificial:** A matriz é enviada para a API da IA, que executa o algoritmo de IA e retorna a coordenada jogada.
3. **Controle CNC:** A coordenada é convertida em comandos G-Code (e.g., G0 X10 Y5) e enviada ao Arduino via serial. O CNC move o eletroímã até a peça, a transporta para a nova posição e retorna à origem.

## 9 Cronograma

The image is a placeholder for a screenshot of a Gantt chart. It contains the text 'images/ScreenshotCronograma1.png' which likely refers to the source of the image. The Gantt chart itself is not visible in this view.

images/ScreenshotCronograma1.png

Figura 3: Diagrama de Gantt



images/Cronograma2.png

Figura 4: Cronograma do Projeto

## Referências

- [1] <https://arxiv.org/abs/2109.11434>
- [2] <https://arxiv.org/abs/2106.11034>
- [3] <https://journals.copmadrid.org/psed/art/psed2025a10>
- [4] <https://pubmed.ncbi.nlm.nih.gov/28494023/>
- [5] MakerC. *DrawingBot: Open Source Plotter*. Thingiverse, 2016. Disponível em: <https://www.thingiverse.com/thing:2348062>.
- [6] Conrado, A. *Controle de Máquinas CNC com GRBL*. Revista de Engenharia, 2017.
- [7] Handson Technology. *CNC Shield V3 Documentation*. 2021. Disponível em: <https://www.handsontec.com>.