

A casa que joga: Um estudo de caso sobre automação residencial em IoT, YOLO e Jogos

The playing house: A case study on home automation in IoT, YOLO, and Games

**Francisco Álisson Vêras Damasceno, Juan David Bizerra Pimentel,
Arthur de Castro Callado, Paulyne Matthews Jucá**

¹Campus de Quixadá - Universidade Federal do Ceará (UFC)
Caixa Postal 5003 – 63.902-580 – Quixadá - CE – Brasil

francisco.ally57@gmail.com, juandbpimentel@alu.ufc.br

[arthur, paulyne]@ufc.br

Abstract. Introduction: *The convergence between Internet of Things (IoT) ecosystems and digital games enables novel interactive experiences. However, real-time integration between game states and physical home automation lacks further exploration. Objective:* This study aims to propose and validate an architecture that integrates YOLO and Home Assistant with digital games, enhancing the player experience through in-home feedback linked to in-game events. A case study methodology was adopted using Genshin Impact as the test case. **Methodology:** *The process involved: (1) configuring Home Assistant with smart lighting; (2) collecting and annotating 1.313 game screenshots for icon detection; (3) training a YOLOv12n model; (4) implementing a state machine for element-color mapping; and (5) evaluating Quality of Experience (QoE) metrics: latency, visual accuracy, and false detection rates. Results:* Results indicate an average system latency of 566 ms, visual accuracy of 82,8% for correct color/state matches, and false positive/false negative rates of 18,35% and 1,85%, respectively.

Keywords *IoT, YOLO, Home Assistant, Games, Quality of Experience.*

Resumo. Introdução: *A convergência entre ecossistemas da Internet das Coisas (IoT) e jogos digitais possibilita experiências interativas diferentes. Contudo, a integração em tempo real entre estados do jogo e automação residencial permanece inexplorada. Objetivos:* Este estudo objetiva propor e validar um modelo aplicável que integre YOLO e Home Assistant com jogos digitais, incrementando a experiência do jogador mediante feedback em sua residência vinculado a eventos no jogo. **Metodologia:** *Adotou-se a metodologia de estudo de caso utilizando Genshin Impact como unidade-caso. O processo envolveu: (1) configuração do Home Assistant; (2) coleta e anotação de 1.313 capturas de tela do jogo para detecção de ícones; (3) treinamento do modelo YOLOv12n; (4) implementação da máquina de estados para mapeamento elemento-cor; e (5) avaliação das métricas de Qualidade de Experiência (QoE): latência do serviço, precisão visual e taxas de detecção falsa. Resultados:* Os resultados indicam uma latência média do sistema de 566 ms, precisão visual de 82,8% para correspondências cor-correta/estado, e taxas de falso positivo/falso negativo de 18,35% e 1,85%, respectivamente.

Palavras-Chave *IoT, YOLO, Home Assistant, Jogos, Qualidade de experiência.*

1. Introdução

A Internet das Coisas (IoT) tem registrado crescente adoção em ambientes residenciais, viabilizando serviços por meio da interconexão de dispositivos, sejam físicos ou virtuais [Mashal et al. 2023]. Esses ecossistemas são tipicamente equipados com dispositivos conectados à *Internet* que controlam funções de iluminação, climatização e entretenimento doméstico [Magrani 2018]. Essa conectividade estabelece uma infraestrutura diversa dentro do lar, madura para interações inovadoras.

Tal conectividade é marcada por avanços em inteligência artificial (IA), realidade virtual (VR) e tecnologias emergentes, possibilitando assim novas formas de interação digital [Zou et al. 2025] ao integrar-se a setores como indústria, saúde, logística, residências e entretenimento [Lara et al. 2021]. Essa integração pode reconfigurar experiências imersivas, especialmente nos jogos digitais, modificando interações digitais e redefinindo as experiências de jogador [Kostadimas et al. 2025].

Os jogos digitais são sistemas em tempo real controlados por jogadores, estabelecendo fluxos bidirecionais com o sistema, em que o jogador envia comandos e recebe *feedback* por meio de atualizações na cena, geralmente representadas por componentes gráficos como ícones [Hougaard e Knoche 2024]. Contudo, quando esses ícones são posicionados em zonas periféricas, isto é, fora da percepção do jogador, podem comprometer a experiência devido à baixa percepção. A fim de auxiliar a percepção do jogador na detecção desses estados do sistema, utilizou-se o modelo de *deep learning* YOLO como algoritmo para visualização dos ícones nesse estudo.

O YOLO (*You Only Look Once*) é um algoritmo de detecção de objetos em tempo real baseado em Redes Neurais Convolucionais (CNNs) que se destaca pelo processamento integral da imagem em uma única etapa [Moksyakov et al. 2024]. A arquitetura trata a detecção como um problema de regressão, enquanto mapeia *pixels* em caixas delimitadoras separadas para rotular objetos no cenário. Aplicado ao ambiente de jogos, o algoritmo atua como um sistema de percepção, que, uma vez identificado, engatilha a lógica do jogo, que aciona dispositivos externos, via *Home Assistant*, para modificar o ambiente físico do jogador. Essa integração viabiliza interações entre os domínios físicos e digital, abrindo novas possibilidades criativas para o *game design*.

Apesar dos avanços individuais na IoT e jogos digitais, a integração de estados de jogo em tempo real e automação residencial permanece uma fronteira inexplorada [Carneiro et al. 2023]. Este artigo busca preencher essa lacuna, propondo uma arquitetura que concilia visão computacional com a automação residencial, especificamente para o *design* de jogos. Assim, a pesquisa está alinhada ao Grande Desafio C2 da área de Jogos e Entretenimento Digital no Brasil (2020-2030), ao focar em como elementos além da tela afetam a interação e alteram a imersão do jogador [Santos e Hounsell 2023] quando integrados a elementos extra-tela.

O artigo estrutura-se da seguinte forma: A Seção 2 apresenta uma revisão bibliográfica; na Seção 3 discutem-se os trabalhos relacionados; a Seção 4 detalha o método seguido, definindo o caso, protocolo e coleta de dados; Os resultados do trabalho desenvolvido estão presentes na Seção 5; As considerações finais foram expostas na Seção 6; Também foi incluída, na Seção 7 a disponibilidade dos artefatos. Por fim, a Seção 8 lista os agradecimentos e detalha o uso de ferramentas de I.A. na confecção deste trabalho.

2. Fundamentação Teórica

A seguir, serão descritas as seções de fundamentação para a construção do projeto, que são: Internet das Coisas, detecção de objetos e jogos digitais.

2.1. Internet das Coisas

A Internet das Coisas (IoT - do inglês, *Internet of Things*) é um paradigma computacional que estabelece redes ubíquas por meio da conexão de dados, pessoas, objetos e aplicações pela *Internet* [dos Santos et al. 2024]. Para isso, a IoT interconecta dispositivos heterogêneos, capacitando-os a desempenhar funções diversas. Seu propósito é permitir que tais objetos não apenas colem, mas também tratem e transfiram dados, convertendo-os em objetos inteligentes, ou seja, passíveis de serem controlados e prover serviços sob controle remoto [Mancini 2017].

A infraestrutura da IoT demanda a integração de objetos inteligentes (como dispositivos *wireless*, sensores e robôs), redes tecnológicas e seres humanos, operando sob diferentes protocolos de comunicação [Eladl 2017]. Tais protocolos permitem que as Coisas (*Things*) se adaptem a diversas situações, contextos, necessidades e modelos de aplicação, viabilizando a interseção da IoT com setores como indústria, saúde, logística e entretenimento [Lara et al. 2021]. Nesse cenário, a convergência tecnológica possibilita o uso de dispositivos de automação como fontes de entrada para aplicações de multimídia, particularmente para a imersão de jogos digitais [Kurihara et al. 2018].

Esta seção apresenta as possibilidades de integração entre a Internet das Coisas e jogos digitais. Referem ainda [Farkas et al. 2024] a relação entre jogos e a conectividade da IoT, na qual ambos os domínios compartilham conceitos de entrada (*input*), processamento e saída (*output*). Nesse contexto, a visão computacional, viabilizada por técnicas de detecção de objetos, atua transformando os fluxos de jogo em dados gatilhos para sistemas em que, uma vez processados, alimentam a lógica de controle que administra tanto as ações do jogo quanto os eventos de saída por objetos inteligentes.

2.2. Detecção de Objetos

As Redes Neurais Convolucionais (CNNs), inspiradas no córtex visual, são utilizadas desde os anos 80 para reconhecimento de imagem [Fukushima 1980]. Tais redes contribuem em aplicações multimídias, como classificação de vídeos, busca por imagem e detecção de objetos em tempo real em jogos [Géron 2022, Asmara et al. 2024].

O YOLO é um algoritmo de detecção de objetos proposto por [Redmon et al. 2016] que se encontra atualmente na versão YOLOv12¹, a qual introduz avanços significativos em relação à proposta inicial [Tian et al. 2025]. O diferencial desta iteração está em combinar CNNs para a identificação de objetos com a *Area Attention Mechanism*, não mais detectando os objetos individualmente, mas em áreas ou grupos semelhantes que estejam próximos [Li et al. 2019]. Essa abordagem busca equilibrar o uso das CNNs ao diminuir a complexidade de operações realizadas. Como resultado, o modelo alcança uma economia no custo computacional sem comprometer a precisão obtida nos modelos anteriores.

¹<https://yolov12.com/>

Com esse avanço, a aplicação em novas áreas de estudo ou melhoria em áreas já existentes se torna a cada dia mais viável, como, por exemplo, em aplicações na saúde, na identificação de órgãos, na colheita de alguns tipos de frutas e leguminosas ou em questões de vigilância [Lemay 2019, Lawal 2021, Ahmed et al. 2021]. Desta forma, esse trabalho propõe expandir as áreas em que o modelo YOLO pode ser aplicado, utilizando-o como um intermediador que integre as detecções a um sistema de IoT para atualizações de estados de dispositivos em tempo real, usando como exemplo um jogo digital.

2.3. Jogos Digitais

Os jogos digitais são softwares que manipulam recursos multimídia (imagens, áudio e vídeo), sendo artefatos com características diferentes de jogos não digitais [Lucchese e Ribeiro 2009]. Esses sistemas podem ser divididos em três partes: enredo, motor e interface interativa. O enredo estabelece a narrativa e o tema; o motor (*engine*) é responsável pelo processamento de entradas (*inputs*) e saídas (*outputs*); e a interface permite a comunicação entre o jogador e o sistema [Battaiola 2000].

A interface atua, por meio de menus e botões, ensinando as regras e o funcionamento do jogo digital ao usuário, implícita ou explicitamente [Saunders e Novak 2013]. Dentre essas regras, destacam-se os modificadores de estado do jogador, aplicados pelo próprio jogador ou por adversários (reais ou imaginários). Esses efeitos podem ser fortalecedores (*buffs*) ou enfraquecedores (*debuffs*); eles são visualmente representados por ícones de *status* em tempo real, servindo como um indicador da interface para o jogador [Denisova e Cook 2019]. A relevância da atenção do jogador com esses indicadores reside na tomada de decisão baseada em informações que alteram sua estratégia imediata. Dessa forma, esses ícones não são meros elementos estéticos, possuindo impacto na percepção do jogo e nas mecânicas [Rau e Ji 2018].

Portanto, no contexto de jogos com sistemas de controle de estados, como o *Genshin Impact*², os *debuffs* constituem mecânicas que, além de indicar as condições do personagem, também influenciam suas estratégias ao provocar reações elementais³, isto é, a combinação entre diferentes efeitos. Assim, esse trabalho propõe-se a investigar o monitoramento desses *status* para o desenvolvimento de sistemas de automação que respondam às condições de jogo.

3. Trabalhos Relacionados

O estudo de [Huang e Ng 2021] visa avaliar o uso de jogos sérios como ferramenta de apoio cultural baseado em tecnologias conectadas. Utilizando um estudo de campo num museu, a metodologia incluiu o desenvolvimento de cinco exposições interativas, como: (1) jogo de trocas, (2) simulador de ourives, (3) *design* de lanterna, (4) moedor interativo e (5) decoração de casa. Os autores concluem que fatores de engajamento e aprendizado foram percebidos como imersivos ao facilitar a compreensão da história aos visitantes. Em suas limitações, o estudo aponta para desafios na criação de ideias para jogos baseados em dispositivos inteligentes, como também aponta para a disponibilidade de plataformas de desenvolvimento voltadas para aplicações residenciais. Assim, este trabalho contribui ao demonstrar o uso da IoT como *design* de inovação, ao permitir a criação de narrativas imersivas por meio de exposições interconectadas aos jogos.

²<https://genshin.hoyoverse.com/>

³<https://shorturl.at/cz6LD>

O estudo de [Asmara et al. 2024], por exemplo, visou à implementação de um *bot* para mira automática utilizando detecção de objetivos via YOLO. O processo deu-se nas seguintes etapas: (1) treinamento do modelo; (2) captura de *frames*; (3) pré-processamento; (4) inferência para detecção; (5) filtragem por Supressão Não-Máxima (NMS); e (6) cálculo das distâncias entre alvos válidos para geração de comandos de movimento do mouse. Como métrica de avaliação, o YOLOv5 se destacou pela acurácia em alguns títulos, como: *Far Cry New Dawn* (92%), *A Sniper Ghost Warrior Contracts 2* (86%), *CS:GO* (76%) e *Point Blank* (57%), com *framerate* de detecção de 35 FPS.

Outro exemplo é encontrado em [Jensn 2018], em que o desenvolvedor implementa um *script* conhecido como *Lifelines*, que controla dispositivos IoT residenciais por meio de retângulos horizontais em capturas de tela. O projeto integra bibliotecas de visão computacional (*cv2* e *pillow*) com a plataforma de automação residencial de código aberto *Home Assistant*. Como estudo de caso, o autor demonstra no jogo *Heroes of the Storm* um vínculo entre os estados do personagem (vida e mana) e lâmpadas RGB. Dessa forma, o projeto funciona como ponto de partida para a proposta seguinte, na hipótese de utilizar o *Home Assistant* com modelos de detecção de objetos, como o YOLO, aplicando-os em outro contexto de jogo, no caso, *Genshin Impact*, a fim de explorar novas possibilidades de interação entre os sistemas de estados e os dispositivos IoT residenciais.

4. Metodologia

O presente trabalho é um estudo quantitativo que propõe a integração entre dispositivos IoT, detecção de objetos e jogos eletrônicos. Para alcançar os resultados almejados, utilizou-se o método estudo de caso. A pesquisa foi orientada a partir da seguinte questão: **Como projetar uma arquitetura utilizando Home Assistant que integre o controle de dispositivos IoT, detecção de objetos e a interação com jogos?**

Conforme proposto por [Gil et al. 2017], cujo referencial metodológico foi delineado, um estudo de caso aborda as seguintes etapas: formulação do problema ou questão de pesquisa; definição das unidades de caso e seleção dos casos; elaboração do protocolo; coleta de dados; análise e interpretação dos dados (Resultados) e relatório (Conclusão). Essas etapas estão ilustradas na Figura 1 abaixo.

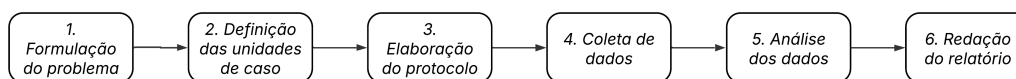


Figura 1. Etapas do protocolo de estudo de caso

4.1. Definição da unidade-caso

Nesse estudo, o contexto do projeto refere-se a um estudo de caso único, que analisa apenas um objeto, especificamente a integração proposta. Partiu-se da premissa de integrar o jogo eletrônico (*Genshin Impact*) com uma lâmpada inteligente utilizando uma plataforma de automação, isto é, *Home Assistant*⁴. Essa integração mudaria de estado com base em *debuffs*⁵ elementais que afetam o jogador. A escolha do jogo justifica-se

⁴Mais informações em: <https://www.home-assistant.io>

⁵São uma modificação temporária nas estatísticas base de um personagem do jogo. *Debuffs* podem apresentar ícones referente ao *status* para transmitir o que está acontecendo [Denisova e Cook 2019].

tanto pela máquina de estado das reações elementais quanto pela colorização de cada elemento. Tais estados são verificados conforme aparecem ou desaparecem da tela do jogador, analisados em tempo real por meio da detecção de objetos YOLO. Ressalta-se que os *status Geo* e *Anemo* não foram utilizados por não terem aparecido no estudo de caso.

4.2. Elaboração do protocolo

Na etapa de elaboração do protocolo, estabelecem-se as tomadas de decisões da pesquisa. Tais decisões foram materializadas mediante provas de conceito (PoC)⁶. Para esse projeto, desenvolveram-se as seguintes PoCs:

1. **Investigação:** Ligar e mudar a cor da lâmpada.
2. **Captação:** Captar imagens para treinar o modelo.
3. **Correção:** Corrigir rótulos.
4. **Treinamento:** Treinar o modelo.
5. **Execução:** Aplicar o modelo treinado em tempo real.

Na etapa de investigação, o Home Assistant foi instalado no computador de um dos pesquisadores. Após a instalação, configuraram-se os dispositivos IoT do projeto: lâmpadas do modelo *Kian Smart Trend Wi-Fi colors LED 11W*. Essas lâmpadas foram integradas com o Home Assistant de duas formas diferentes em experimentos preliminares. A primeira utilizou o serviço de nuvem da plataforma, gerando problemas de latência, com atraso significativo entre um de ligar/desligar e seu efeito. No segundo método, foi empregado o *plugin Local Tuya*⁷, que opera as lâmpadas em rede local. Esta opção foi adotada por apresentar uma latência bem menor em relação à primeira.

Na etapa de captura, implementou-se um *script* Python, que, além de coletar as imagens em formato JPEG, também gerava um arquivo de texto contendo o rótulo identificando o elemento capturado e as coordenadas das caixas delimitadoras⁸. O programa operava em segundo plano durante a execução do jogo, acionado por atalhos para auxiliar o pré-processamento das imagens destinadas ao treinamento do modelo. Foram coletadas 1313 imagens, divididas entre 5 classes elementais. A captura ocorreu em 5 de julho de 2025, das 20h21 às 20h28. No entanto, observou-se que a captura gerava arquivos com seis caixas delimitadoras e um único rótulo padrão, demandando ajustes.

Na correção dos rótulos, as imagens foram organizadas em diretórios nomeados conforme seus respectivos rótulos. Esse processo permitiu identificar os elementos presentes em cada imagem e ajustar seus rótulos. Dessa maneira, caixas delimitadoras sem elementos foram eliminadas, enquanto rótulos incorretos foram corrigidos para seus respectivos rótulos conforme a nomenclatura da pasta, tornando o processo de rotulagem mais ágil. Por exemplo, arquivos contidos na pasta *Pyro* tiveram seus rótulos mapeados para o identificador 0, correspondente à classe desse elemento.

Na etapa de treinamento, organizou-se a pasta *dataset* conforme as especificações da biblioteca do YOLO. Dentro dessa pasta, havia a pasta “imagens”, que armazenava

⁶Uma prova de conceito é um passo inicial para testar e avaliar uma ideia conceitual, de modo a obter uma noção de como uma ideia será executada na prática [SILVA e COSTA 2019].

⁷É possível encontrar a documentação em: <https://codetheweb.github.io/tuyapi/>

⁸Do inglês *boundary boxes*, que representam o tamanho e a posição precisa do objeto.

todas as imagens coletadas, e a pasta *labels*, contendo os arquivos de anotação com os rótulos e as caixas delimitadoras. Em seguida, realizou-se uma partição dos dados em proporções: treino (60%), validação (10%) e teste (30%). Essa divisão é necessária para o ajuste dos hiperparâmetros durante o treinamento do modelo sem vieses e a análise das métricas de desempenho do modelo no conjunto de teste após o treino. O processo teve um tempo total de treinamento de aproximadamente 3 horas e 8 minutos.

Na etapa de execução, foi utilizado um *script* que captura em vídeo em tempo real da tela. Durante a captura, as imagens são submetidas à inferência do modelo treinado anteriormente, o qual é integrado a uma máquina de estados responsável por verificar o elemento e sua hierarquia e também fazer a comunicação com o Home Assistant, controlando as lâmpadas inteligentes.

Dessa forma, a Figura 2 ilustra a arquitetura elaborada. Depois de tirar a captura de tela, os *frames* do jogo são varridos para que o YOLO possa encontrar os ícones. Em seguida, as inferências são transferidas para a máquina de estados, que determina o que fazer com a detecção. Caso ocorra uma alteração de estado da máquina, uma requisição é disparada para a plataforma de Home Assistant, que envia o comando para a lâmpada inteligente mudar a cor segundo a cor definida para o estado.

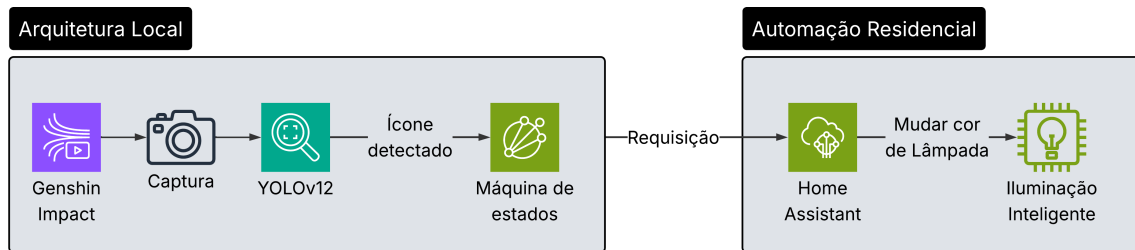


Figura 2. Arquitetura proposta

4.3. Coleta de dados

Na etapa de coleta dos dados, analisaram-se três métricas de Qualidade de Experiência (QoE). As métricas coletadas foram a latência do serviço percebida, precisão visual e frequência de detecções falsas. A coleta ocorreu em 15 de julho de 2025, com duração aproximada de três horas.

Para a métrica de **latência do serviço percebida**, foi realizada a medição do atraso entre o aparecimento do *debuff* elemental na tela e a reação de troca de cor da lâmpada. Para tanto, gravou-se simultaneamente a tela do jogo e a lâmpada com auxílio de uma *webcam*.

Na **frequência de detecções falsas**, ocorreram sessões de jogo estendidas para medir ocorrências de eventos (viagem rápida e combate com múltiplos elementais), registrando-se se a lâmpada reagiria sem um ícone na tela (falsos positivos) ou falharia em reagir quando o ícone estivesse presente (falsos negativos). As taxas foram quantificadas pela Taxa de Falsos Positivos ($TFP = \frac{FP}{FP+VN} \times 100\%$) e pela Taxa de Falsos Negativos ($TFN = \frac{FN}{FN+VP} \times 100\%$)⁹.

⁹Em que **FP** representa falsos positivos (lâmpada reage e sem *debuff*), **VN** verdadeiros negativos (lâmpada não reage e sem *debuff*), **FN** falsos negativos (lâmpada não reage e com *debuff*) e **VP** verdadeiros positivos (lâmpada reage e com *debuff*).

Na avaliação da **precisão visual**, mediu-se o quão precisas foram as correspondências entre os ícones na tela e a cor da lâmpada, categorizando as respostas como: cor correta, incorreta ou não acendeu. Para tal, definiram-se as cores esperadas para cada ícone. A partir dessa definição, foi elaborada uma matriz de estados para determinar as cores de saída, transições e prioridades dos *debuffs*, conforme ilustrado na Tabela 1 abaixo.

Tabela 1. Matriz de Estados

	Hydro	Cryo	Electro	Pyro	Dendro	Sem Estado
Hydro	—	Cryo	Electro	—	—	Sem Estado
Cryo	Hydro	—	Electro	—	—	Sem Estado
Electro	Hydro	Cryo	—	Pyro	—	Sem Estado
Pyro	—	—	Electro	—	Dendro	Sem Estado
Dendro	—	—	—	Pyro	—	Sem Estado
Sem Estado	Hydro	Cryo	Electro	Pyro	Dendro	Sem Estado

5. Resultados

Como resultados obtidos, teve-se a captura de *timestamps* entre os *frames* de aparecimento dos ícones e as reações de troca de cor da lâmpada, totalizando 88 *timestamps* para uma amostra significativa. Ressalta-se que alguns *frames* são mais precisos em comparação a outros, visto que o ícone pode surgir de duas maneiras diferentes no jogo: gradual ou instantânea. Assim, a latência de serviço obteve um tempo médio de resposta de aproximadamente **566 milissegundos**.

Embora perceptível ao usuário, essa latência pode ser considerada aceitável, dado que diferentes gêneros de jogos apresentam tolerâncias distintas à latência [Zha e Zhang 2019]. No contexto de RPGs, gênero ao qual *Genshin Impact* pertence, cenários de combate podem afetar a experiência planejada devido ao atraso, por se tratarem de situações que requerem agilidade. Porém, em cenários de exploração, onde o atraso não é crítico, infere-se que a latência percebida não comprometeria a qualidade da experiência do usuário.

Outra métrica avaliada foi a precisão visual, contabilizada quando a cor era emitida pela lâmpada, enquanto o ícone correspondente estava em tela. Foram coletadas 126 alterações de estado da lâmpada, com as correspondências entre os ícones e a reação da lâmpada, além de dois casos anômalos. As correspondências corretas representaram ($n = 106$) dos casos, enquanto cores incorretas foram ($n = 20$), que ocorreram quando a lâmpada reagiu sem a presença dos ícones na tela. Houve também dois casos anômalos, em que o ícone estava na tela do jogo, porém a lâmpada não mudava de cor e nem reagia.

Após a coleta de 20 ocorrências de cores incorretas, mapearam-se os cenários em que a lâmpada reagia sem correspondência com o ícone exibido em tela, incluindo movimentação do personagem, acesso a menus e situações de combate. O elemento *Electro* registrou a maior frequência de erros (11), enquanto *Hydro* e *Pyro* apresentaram duas ocorrências cada. O elemento Dendro foi detectado principalmente em cenários

de movimentação (4) e exploração (1). Notou-se também que o elemento *Cryo* não foi observado nessas inconsistências, embora tenha sido registrado em uma das ocorrências anômalas citadas anteriormente.

Ao tirar as frequências de uma matriz de confusão (Figura 3a), levando em conta a presença de um *debuff* na tela e a detecção correta dele como verdadeiro positivo, registraram-se 106 ocorrências de verdadeiros positivos (presença do *debuff* com reação luminosa) contra 89 verdadeiros negativos (ausência do *debuff* sem reação luminosa). Nota-se um desequilíbrio na amostra, uma vez que a lâmpada foi acionada mais vezes do que desligada. Isso deve-se ao mecanismo de transição entre os elementos: em vez de retornar ao estado neutro (ausência de *debuff*) antes de ativar um novo elemento, o sistema alterna diretamente entre estados ativos. Por exemplo, em dado momento o estado do elemento *Pyro* é trocado diretamente para *Hydro*, sem passar pelo estado neutro (lâmpada inativa). Essa situação reduz frequências de observação do estado neutro, tornando consequentemente a contagem de verdadeiros positivos maior que a de verdadeiros negativos.

Enquanto, entre as métricas de desempenho (Figura 3b), alcançou em destaque o Recall em 98% (0.98), em contraste ao F1-score com 90% (0.90) e à Precisão com 84% (0.84). Na prática, o modelo apresenta uma performance adequada em identificar os positivos reais, mas comete uma quantidade considerável de alarmes falsos, por conta de sua baixa precisão em relação às outras métricas. Esse *trade-off* é evidente nos valores de F1-score, sendo uma métrica crítica para tarefas de classificação dos tipos elementares envolvendo dados dinâmicos pertencentes ao domínio dos jogos digitais.

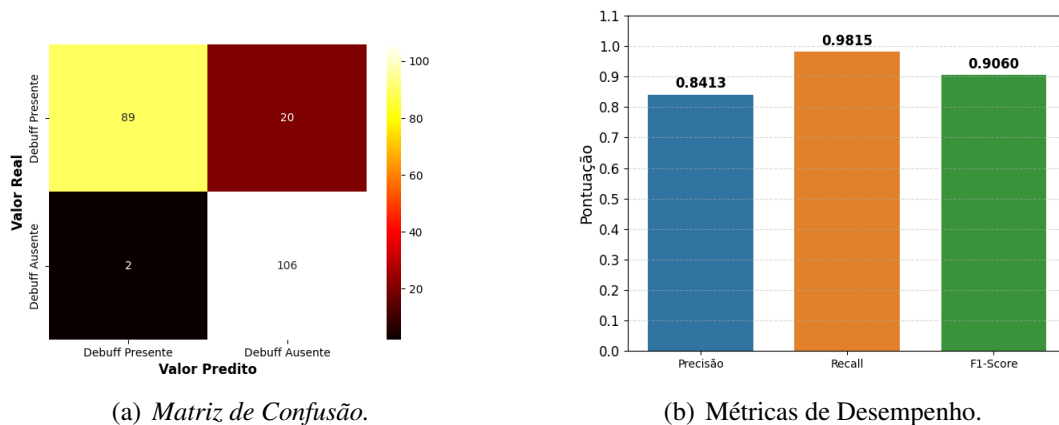


Figura 3. Avaliação de desempenho do modelo.

A análise das detecções revelou que a Taxa de Falsos Positivos foi de $(\frac{20}{20+89} \times 100\% = 18,35\%)$, indicando que o modelo ocasionalmente interpreta elementos visuais do ambiente ou da interface como ícones de *debuff*. Assim, em um quinto das vezes, a lâmpada reage sem que haja um ícone correspondente na tela. Por outro lado, a Taxa de Falsos Negativos foi de apenas $(\frac{2}{106+2} \times 100\% = 1,85\%)$, o que demonstra a eficácia do sistema em identificar a presença real dos ícones, com falhas de omissão consideradas mínimas. Esses resultados indicam que, embora o sistema seja sensível a falsos negativos, ele ainda apresenta uma detecção considerada bastante eficaz ao acionar a lâmpada corretamente em cenários divergentes.

6. Considerações Finais e Trabalhos Futuros

Este estudo apresentou um modelo de integração entre visão computacional (YOLO), Home Assistant e jogos digitais. A premissa consistiu em integrar o jogo *Genshin Impact* como fonte de dados, alterando a iluminação residencial conforme o ícone detectado pelo modelo. Os resultados indicaram uma latência média de 566 ms, valor considerado aceitável para a experiência do usuário. Observou-se, contudo, que em cenários de combate e exploração há influência na precisão das cores, com a ocorrência de falsos positivos, isto é, sem reação luminosa em 18,35% dos casos devido à confusão do modelo com os elementos do ambiente e os ícones elementais.

Como principal limitação encontrada, identificou-se uma anomalia na detecção do elemento *Cryo*. Que, apesar de o ícone estar visível em tela, o sistema por vezes falha em disparar a automação, sugerindo que o conjunto de dados de treinamento para este elemento carece de diversidade. Assim, tal achado infere que as imagens utilizadas no treinamento do modelo não estão adequadas para esse elemento, indicando a necessidade de um re-treinamento específico.

Quanto às contribuições desta pesquisa, residem na exploração da interseção entre IoT e entretenimento, demonstrando como o ambiente ao redor do jogador pode reagir aos estados lógicos do jogo. E como trabalhos futuros, pretende-se explorar integrações temáticas, como a conexão entre o jogo *Frostpunk* e condicionadores de ar inteligentes, utilizando o controle térmico como fator de imersão para simular o clima gélido da narrativa. Por fim, planeja-se expandir a pesquisa para testes com usuários reais, permitindo analisar o impacto sobre a experiência dessa tecnologia na experiência de jogo-jogador.

7. Disponibilidade dos Artefatos

Todos os artefatos, incluindo o conjunto de dados, *scripts* e um vídeo ilustrativo de parte da captura deste estudo, estão disponíveis em:

- (i) Dados: <https://doi.org/10.6084/m9.figshare.31164331>
- (ii) Código: <https://github.com/AcademicJPAV/IoTGameColors>
- (iii) Vídeo: <https://youtu.be/JK4kcH8SktI?si=V8wnioTo4Ov-hVka>

8. Agradecimentos

Este trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Os autores também gostariam de agradecer à Universidade Federal do Ceará - Campus Quixadá (UFC) pelo apoio educacional.

Neste trabalho não foram utilizadas ferramentas de Inteligência Artificial generativa para produção ou revisão de conteúdo acadêmico. Porém, para apoio à escrita, os autores utilizaram a ferramenta *LanguageTool*¹⁰. Todo o conteúdo científico produzido, incluindo análise, interpretação e resultados, é de responsabilidade integral dos autores.

¹⁰O *LanguageTool* é uma ferramenta para verificação gramatical e estilística. Mais informações em: <https://languagetool.org/pt-BR/>

Referências

- Ahmed, I., Ahmad, M., Rodrigues, J. J., Jeon, G., e Din, S. (2021). A deep learning-based social distance monitoring framework for covid-19. *Sustainable cities and society*, 65:102571.
- Asmara, R. A., Anugrah, M. R. S., Wibowo, D. W., Arai, K., Burhanuddin, M. A., Handayani, A. N., e Damayanti, F. A. (2024). Yolo-based object detection performance evaluation for automatic target aimbot in first-person shooter games. *Bulletin of Electrical Engineering and Informatics*, 13(4):2456–2470.
- Battaiola, A. L. (2000). Jogos por computador–histórico, relevância tecnológica e mercadológica, tendências e técnicas de implementação. *Anais do XIX Jornada de Atualização em Informática*, pages 83–122.
- Carneiro, N., Nunes, C., Raulino, N., Gomes, I., Andrade, R., e Darin, T. (2023). Iot games and gamified systems: Summertime sadness or lust for hype? In *Proceedings of the XXII Brazilian Symposium on Human Factors in Computing Systems*, pages 1–12.
- Denisova, A. e Cook, E. (2019). Power-ups in digital games: The rewarding effect of phantom game elements on player experience. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '19, page 161–168, New York, NY, USA. Association for Computing Machinery.
- dos Santos, M. R., de Castro Callado, A., e Laurindo, L. E. C. (2024). Enhancing internet of things conceptual, practical, and programming learning skills. In *Workshop sobre Educação em Computação (WEI)*, pages 431–442. SBC.
- Eladl, G. H. (2017). Technical requirements for the application of internet of things. *International. Journal of Computer Science and Network*, 6(4).
- Farkas, T., Hughes, N. G. J., e Fiebrink, R. (2024). How boardgame players imagine interacting with technology. *Proceedings of the ACM on Human-Computer Interaction*, 8(CHI PLAY):1–24.
- Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202.
- Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. "O'Reilly Media, Inc."
- Gil, A. C. et al. (2017). *Como elaborar projetos de pesquisa*, volume 4. Atlas São Paulo.
- Hougaard, B. I. e Knoche, H. (2024). Aiming, pointing, steering: A core task analysis framework for gameplay. *Proceedings of the ACM on Human-Computer Interaction*, 8(CHI PLAY):1–48.
- Huang, H. e Ng, K. (2021). Designing for cultural learning and reflection using iot serious game approach. *Personal and Ubiquitous Computing*, 25:509–524.
- Jensn, J. (2018). lifelights: Video games blended with home automation. <https://github.com/jjensn/lifelights>. Acessado em: 26/04/2026.

- Kostadimas, D., Kasapakis, V., e Kotis, K. (2025). A systematic review on the combination of vr, iot and ai technologies, and their integration in applications. *Future Internet*, 17(4).
- Kurihara, K. et al. (2018). Gamecontrollerizer: Middleware to program inputs for augmenting digital games. *arXiv preprint arXiv:1810.01070*.
- Lara, J. E., Reis, L. J., Tissot-Lara, T. A., e Silva, A. O. (2021). Admirável mundo novo na perspectiva da tríade: Internet das coisas, pessoas e mercados. *Perspectivas em Ciência da Informação*, 26(02):124–150.
- Lawal, M. (2021). Tomato detection based on modified yolov3 framework, sci. rep., 1447.
- Lemay, A. (2019). Kidney recognition in ct using yolov3. *arXiv preprint arXiv:1910.01268*.
- Li, Y., Kaiser, L., Bengio, S., e Si, S. (2019). Area attention. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3846–3855. PMLR.
- Lucchese, F. e Ribeiro, B. (2009). Conceituação de jogos digitais. *São Paulo*, 7.
- Magrani, E. (2018). *INTERNET DAS COISAS, A ED. 1*. Editora FGV.
- Mancini, M. (2017). Internet das coisas: História, conceitos, aplicações e desafios. *Project Management Institute–PMI*.
- Mashal, I., Shuhaiber, A., e Al-Khatib, A. W. (2023). User acceptance and adoption of smart homes: A decade long systematic literature review. *International Journal of Data and Network Science*, 7(2):533.
- Moksyakov, A., Wu, Y., Gadsden, S. A., Yawney, J., e AlShabi, M. (2024). Object detection and tracking with yolo and the sliding innovation filter. *Sensors*, 24(7).
- Rau, P.-L. P. e Ji, Y. G. (2018). Games and hci. *International Journal of Human–Computer Interaction*, 34(8):681–681.
- Redmon, J., Divvala, S., Girshick, R., e Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.
- Santos, R. P. D. e Hounsell, M. D. S. (2023). *Grand Research Challenges in Games and Entertainment Computing in Brazil–GrandGamesBR 2020–2030*. Springer.
- Saunders, K. e Novak, J. (2013). *Game Development Essentials: Game interface design*. Delmar, Cengage Learning.
- SILVA, C. B. d. e COSTA, M. (2019). Prova de conceito (poc): o que é e como construir? *Blog NectaCRM*.
- Tian, Y., Ye, Q., e Doermann, D. (2025). Yolov12: Attention-centric real-time object detectors. *arXiv preprint arXiv:2502.12524*.
- Zha, M. e Zhang, Y. (2019). *The Effects of Network Latency on Player Gaming Experience*. PhD thesis, Worcester Polytechnic Institute.

Zou, Y., Kuek, F., Feng, W., e Cheng, X. (2025). Digital learning in the 21st century: trends, challenges, and innovations in technology integration. *Frontiers in Education*, Volume 10 - 2025.