REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

AVALIAÇÃO DE MODELOS DE DETECÇÃO DE OBJETOS PARA VISÃO COMPUTACIONAL ASSISTIVA EMBARCADA

EVALUATION OF OBJECT DETECTION MODELS FOR EMBEDDED ASSISTIVE COMPUTER VISION

EVALUACIÓN DE MODELOS DE DETECCIÓN DE OBJETOS PARA VISIÓN POR COMPUTADORA ASISTIVA EMBEBIDA

Rafaela Laryssa Mello Neto
Estudante de Análise de Sistemas, Instituto Federal de São Paulo (Campinas), Brasil
E-mail: l.rafaela@aluno.ifsp.edu.br / ORCID: 0009-0002-9908-8067

Sophia Ferreira Boonen
Estudante de Análise de Sistemas, Instituto Federal de São Paulo (Campinas), Brasil
E-mail: s.boonen@aluno.ifsp.edu.br / ORCID: 0009-0002-3849-028X

Glauber da Rocha Balthazar
Doutor em Engenharia de Sistemas Agrícolas, Instituto Federal de São Paulo (Campinas), Brasil
E-mail: glauber.balthazar@ifsp.edu.br / ORCID: 0000-0002-1993-6621

Resumo

A deficiência visual pode comprometer a mobilidade e a segurança em ambientes dinâmicos, motivando o desenvolvimento de soluções assistivas capazes de identificar possíveis obstáculos. Este estudo investiga a hipótese de que modelos de visão computacional treinados com dados específicos do cenário de aplicação podem detectar objetos potencialmente relevantes para a navegação sob diferentes condições ambientais, mantendo características adequadas para sistemas embarcados. Um protótipo vestível baseado em Raspberry Pi foi desenvolvido, e diferentes arquiteturas de detecção de objetos e segmentação semântica foram treinadas e avaliadas utilizando um conjunto de dados personalizado composto por cinco classes de objetos. Os modelos YOLO11n e YOLO26 foram selecionados para avaliação experimental em um ambiente conhecido e um ambiente não observado. O YOLO26 alcançou valores mais elevados de mAP@50-95 e níveis de confiança para classes específicas, enquanto o YOLO11n detectou todas as cinco classes no ambiente conhecido. No ambiente não observado, ambos os modelos detectaram três classes. Os resultados demonstram o potencial da abordagem, ao mesmo tempo que revelam limitações de generalização, indicando a necessidade de maior diversidade no conjunto de dados e de avaliações de desempenho em sistemas embarcados.

Palavras-chave: Tecnologia Assistiva; Detecção de Objetos; Visão Computacional Embarcada; YOLO.

Abstract

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Visual impairment can compromise mobility and safety in dynamic environments, driving the development of assistive solutions capable of identifying potential obstacles. This study investigates the hypothesis that computer vision models trained on data specific to the application scenario can detect objects relevant to navigation across varying environmental conditions, while maintaining characteristics suitable for embedded systems. A wearable prototype based on Raspberry Pi was developed, and various object detection and semantic segmentation architectures were trained and evaluated using a custom dataset comprising five object classes. The YOLO11n and YOLO26 models were selected for experimental evaluation in both a known environment and an unseen environment. YOLO26 achieved higher mAP@50-95 values and confidence levels for specific classes, whereas YOLO11n detected all five classes in the known environment. In the unseen environment, both models detected three classes. The results demonstrate the potential of this approach while revealing generalization limitations, highlighting the need for greater dataset diversity and performance evaluations on embedded systems.
Keywords: Assistive Technology; Object Detection; Embedded Computer Vision; YOLO.

Resumen

La discapacidad visual puede comprometer la movilidad y la seguridad en entornos dinámicos, lo que motiva el desarrollo de soluciones de asistencia capaces de identificar posibles obstáculos. Este estudio investiga la hipótesis de que los modelos de visión por computadora entrenados con datos específicos del escenario de aplicación pueden detectar objetos potencialmente relevantes para la navegación bajo diversas condiciones ambientales, manteniendo características adecuadas para sistemas embebidos. Se desarrolló un prototipo vestible basado en Raspberry Pi y se entrenaron y evaluaron diferentes arquitecturas de detección de objetos y segmentación semántica utilizando un conjunto de datos personalizado compuesto por cinco clases de objetos. Los modelos YOLO11n y YOLO26 fueron seleccionados para su evaluación experimental en un entorno conocido y en un entorno no observado. El modelo YOLO26 alcanzó valores más elevados de mAP@50-95 y niveles de confianza para clases específicas, mientras que el YOLO11n detectó las cinco clases en el entorno conocido. En el entorno no observado, ambos modelos detectaron tres clases. Los resultados demuestran el potencial de este enfoque, al tiempo que revelan limitaciones de generalización, lo que indica la necesidad de una mayor diversidad en el conjunto de datos y de evaluaciones de rendimiento en sistemas embebidos.
Palabras clave: Tecnología de asistencia; Detección de objetos; Visión artificial embebida; YOLO.

1. Introdução

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

A deficiência visual pode comprometer a autonomia e a mobilidade dos indivíduos, especialmente em ambientes dinâmicos nos quais obstáculos temporários dificultam a percepção segura do espaço. No contexto escolar, corredores constituem áreas de circulação sujeitas à presença de objetos deslocados, mochilas e mobiliário, que podem interferir no deslocamento dos usuários. A literatura apresenta investigações sobre lesões e comportamentos de risco entre jovens (Pickett et al., 2002), bem como sobre fatores associados à ocorrência de lesões não intencionais no ambiente escolar (Stadtmüller et al., 2022). Esses estudos contribuem para contextualizar a importância da segurança nos espaços escolares, embora não constituam evidências específicas sobre o desempenho de sistemas de detecção de obstáculos para pessoas com deficiência visual.

Avanços em Inteligência Artificial e Visão Computacional têm possibilitado o desenvolvimento de tecnologias assistivas capazes de realizar detecção de objetos em tempo real. Arquiteturas da família YOLO têm sido empregadas nesse contexto devido à relação entre precisão e velocidade de inferência, característica relevante para aplicações que demandam resposta imediata. Revisões da literatura demonstram a crescente utilização dessas técnicas em sistemas destinados à assistência de pessoas com deficiência visual (Valipoor, Antonio, 2023; Klann et al., 2024).

Diferentes abordagens têm explorado modelos de detecção de objetos para navegação assistiva. Alsultan e Mohammad (2023) empregaram YOLOv7 na identificação de objetos em diferentes cenários, enquanto compararam arquiteturas de detecção aplicadas à identificação de obstáculos. Litoriya et al. (2023) e Santiago et al. (2023) demonstraram a viabilidade da execução de modelos de detecção em dispositivos móveis, e Gassenn et al. (2024) combinaram detecção de objetos e estimativa de profundidade para interpretação espacial do ambiente. Soluções vestíveis também têm sido investigadas, como o sistema de Silva e Silva (2025) que associa visão computacional à geração de alertas sonoros. Adicionalmente, Li et al. (2023) demonstraram a importância da utilização de conjuntos de dados específicos do domínio para melhorar o desempenho dos

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

modelos em cenários reais.

Apesar desses avanços, aplicações assistivas embarcadas apresentam desafios relacionados tanto à capacidade de detecção quanto ao custo computacional e à generalização dos modelos. Arquiteturas com maior complexidade podem apresentar desempenho preditivo elevado, porém sua utilização em plataformas com recursos limitados pode comprometer a latência e a taxa de processamento necessárias para aplicações em tempo real. Além disso, o desempenho obtido durante o treinamento pode não se manter diante de variações nos objetos e nos ambientes encontrados durante a utilização do sistema. Dessa forma, torna-se relevante não apenas utilizar dados específicos do domínio, mas também avaliar o comportamento dos modelos em cenários distintos daqueles representados durante a construção do conjunto de dados.

Diante desse cenário, este trabalho propõe o desenvolvimento e a avaliação de um protótipo de sistema assistivo vestível, baseado em visão computacional embarcada, para identificação de objetos potencialmente relevantes para a navegação em corredores escolares. Para isso, foi construído um conjunto de dados específico contendo objetos associados ao contexto de aplicação e foram avaliadas diferentes arquiteturas de visão computacional, considerando métricas de desempenho preditivo e complexidade computacional. A partir dessa análise, os modelos com melhor desempenho foram selecionados para avaliação experimental em corredores escolares, considerando tanto um ambiente conhecido quanto um ambiente não observado nessa etapa.

Portanto, a hipótese investigada neste trabalho é que modelos treinados com dados específicos do ambiente escolar podem identificar objetos potencialmente relevantes para a navegação mesmo quando submetidos a variações no cenário de aplicação, mantendo características computacionais compatíveis com sua utilização em uma plataforma embarcada. Dessa forma, o trabalho busca avaliar tanto a capacidade de identificação dos objetos e o comportamento dos modelos diante de mudanças no ambiente quanto sua aplicabilidade em um dispositivo vestível de prototipação rápida.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

2. Material e Métodos

Este trabalho caracteriza-se como uma pesquisa aplicada e experimental, conduzida por meio do desenvolvimento e da avaliação técnica preliminar de uma prova de conceito (Proof of Concept – PoC) de visão computacional embarcada, concebida para aplicações assistivas. O escopo experimental concentrou-se na construção de um conjunto de dados restrito a cinco classes de objetos presentes em corredores escolares, no treinamento e na avaliação de diferentes arquiteturas de visão computacional, na implementação de um protótipo vestível e na realização de testes exploratórios de detecção em dois ambientes escolares. Dessa forma, a investigação teve como finalidade examinar a viabilidade técnica preliminar da abordagem proposta, sem pretender demonstrar a capacidade de navegação autônoma, a segurança operacional ou a efetividade funcional do dispositivo como tecnologia assistiva. O procedimento metodológico foi organizado em cinco etapas: construção e preparação do conjunto de dados, treinamento das arquiteturas, avaliação e seleção dos modelos, desenvolvimento do protótipo embarcado e avaliação experimental nos ambientes escolares.

Construção e preparação do conjunto de dados

Foram definidas cinco classes de objetos presentes em corredores escolares: *cleaning cart* (carrinho de limpeza), *fire extinguisher* (extintor), *safety* cone (cone de sinalização), *water fountain* (bebedouro) e *trash can* (lixeira). As imagens correspondentes a essas classes foram organizadas na plataforma *Roboflow*, na qual foi realizado o processo de anotação dos objetos. Para as tarefas de detecção, as regiões correspondentes aos objetos foram delimitadas por *bounding* boxes e associadas às respectivas classes. Para os experimentos envolvendo segmentação, o conjunto de dados foi preparado no formato correspondente às máscaras de segmentação.

A seleção das cinco classes de objetos foi fundamentada em critérios relacionados às características dos ambientes escolares e à finalidade assistiva da aplicação proposta. A definição das categorias contou com uma consulta qualitativa

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

a uma estudante com deficiência visual da própria instituição, cuja contribuição auxiliou na identificação de elementos considerados relevantes para a circulação em corredores escolares. Foram considerados três critérios principais: a frequência de ocorrência dos objetos nesses ambientes, sua variabilidade espacial e sua diversidade geométrica. O primeiro critério contemplou elementos presentes nas rotas de circulação da instituição; o segundo buscou representar diferentes condições de posicionamento, incluindo objetos móveis ou próximos ao piso, como carrinhos de limpeza, cones de sinalização e lixeiras, e elementos fixos ou suspensos, como extintores e bebedouros; o terceiro considerou a diversidade de formatos, dimensões e características visuais, permitindo investigar o comportamento dos modelos diante de diferentes configurações de objetos. Dessa forma, as cinco classes constituíram um conjunto inicial de categorias potencialmente relevantes para a navegação em corredores escolares, definido para a avaliação técnica preliminar dos modelos de visão computacional. Ressalta-se que essa seleção não pretende representar a totalidade dos elementos encontrados nesses ambientes, podendo ser ampliada em estudos futuros.

O conjunto de dados inicial foi constituído por 224 imagens reais, obtidas no ambiente escolar e organizadas na plataforma *Roboflow*. Após a divisão dos dados em conjuntos de treinamento, validação e teste, foram aplicadas técnicas de aumento de dados (*data augmentation*) exclusivamente às imagens destinadas ao treinamento. As transformações compreenderam espelhamento horizontal (horizontal flip), responsável pela inversão da orientação horizontal das imagens; rotações aleatórias no intervalo de −10° a +10°, destinadas a representar pequenas variações de posicionamento da câmera; e ajustes de brilho entre −15% e +15%, utilizados para simular alterações nas condições de iluminação do ambiente. Esse procedimento resultou em um conjunto final de 316 imagens, considerando as amostras originais e as versões artificialmente modificadas. Os conjuntos de validação e teste permaneceram compostos exclusivamente por imagens originais, sem aplicação de técnicas de aumento de dados.

O conjunto de dados original foi contemplando com cinco classes de objetos presentes no ambiente escolar: carrinho de limpeza (*cleaning cart*), extintor de

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

incêndio (*fire extinguisher*), cone de sinalização (*safety cone*), bebedouro (*water fountain*) e lixeira (*trash can*). Segundo os registros de organização do conjunto de dados, buscou-se uma distribuição aproximadamente equilibrada entre as categorias. Para fins descritivos, uma distribuição uniforme corresponde a aproximadamente 45 instâncias por classe (uma instância por imagem).

Treinamento das arquiteturas

O treinamento dos modelos foi realizado no ambiente Google Colab. Foram incluídas no protocolo experimental as arquiteturas *YOLO11n*, *YOLO26*, *RT-DETR*, *SSDLite320 MobileNetV3-Large*, *SegFormer* e U-*Net*, contemplando abordagens de detecção de objetos e segmentação semântica. Os procedimentos de treinamento das arquiteturas foram documentados em notebooks desenvolvidos no ambiente *Google Colab*, sendo preservados parcialmente os registros de configuração, os resultados experimentais e os arquivos de pesos dos modelos. A documentação disponível foi utilizada para caracterizar os experimentos e identificar as limitações relacionadas à recuperação integral dos resultados.

Para a arquitetura YOLO11n (ULTRALYTICS, 2024), foi utilizada a variante YOLO11nn (Nano), caracterizada por aproximadamente 2,59 milhões de parâmetros e complexidade computacional de 6,4 GFLOPs, conforme os registros de configuração do modelo. O treinamento foi inicializado a partir do arquivo de pesos pré-treinados `YOLO11nn.pt`, utilizando a implementação disponibilizada pela biblioteca *Ultralytics*. A escolha dessa variante considerou sua menor complexidade computacional em relação às versões de maior porte da mesma família, característica relevante para a investigação de aplicações de visão computacional embarcada. Para a arquitetura YOLO26 (ULTRALYTICS, 2026), foi utilizada a variante YOLO26n (Nano), implementada por meio da biblioteca *Ultralytics*. O treinamento foi inicializado a partir do arquivo de pesos pré-treinados `yolo26n.pt`, conforme registrado no notebook de implementação. A escolha dessa variante considerou sua proposta de menor complexidade computacional em relação às versões de maior porte da mesma família, característica relevante para aplicações de detecção de objetos em dispositivos embarcados com recursos computacionais

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

limitados.

Entre as arquiteturas de detecção de objetos, foi empregada a configuração SSDLite320 com *backbone* SSDLite320 MobileNetV3-Large-Large, implementada por meio da biblioteca *Torchvision*, integrante do framework *PyTorch*. Nessa configuração, a rede SSDLite320 MobileNetV3-Large-Large atua como extratora de características visuais, enquanto a estrutura *SSDLite* realiza a predição das classes e das caixas delimitadoras dos objetos. A arquitetura foi incluída na avaliação preliminar por apresentar uma proposta de detecção com complexidade computacional reduzida, adequada à investigação de soluções para dispositivos com recursos limitados.

O conjunto de dados foi organizado na plataforma *Roboflow* e dividido em três subconjuntos, correspondentes a 70% das imagens para treinamento, 20% para validação e 10% para teste. Os quatro modelos de detecção de objetos (YOLO11n, YOLO26, RT-DETR e SSDLite320 MobileNetV3-Large) utilizaram exatamente as mesmas imagens em cada uma dessas partições, assegurando condições equivalentes quanto à composição dos dados empregados na comparação. Para as arquiteturas de segmentação semântica (*SegFormer* e U-Net), foram utilizadas as representações de dados apropriadas à respectiva tarefa, mantendo-se as proporções de divisão informadas. Considerando as 224 imagens originais, essas proporções correspondem, aproximadamente, a 157 imagens para treinamento, 45 para validação e 22 para teste, antes da aplicação das técnicas de aumento de dados (*data augmentation*), restritas ao subconjunto de treinamento. Após esse procedimento, o conjunto final passou a compreender 316 imagens. A distribuição específica das imagens e instâncias por classe em cada partição não pôde ser recuperada dos registros disponíveis, constituindo uma limitação da documentação do conjunto de dados. A Tabela 1 apresenta a distribuição aproximada das imagens originais entre as três partições.

Tabela 1. Distribuição aproximada das imagens originais entre os subconjuntos de treinamento, validação e teste.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Partição	Proporção	Imagens aproximadas
Treinamento	70%	157
Validação	20%	45
Teste	10%	22
Total	100%	224

Fonte: autores

Como medida para reduzir o risco de vazamento de dados (*data leakage*), a divisão do conjunto original em treinamento, validação e teste foi realizada antes da aplicação das técnicas de aumento de dados. Dessa forma, as transformações artificiais foram restritas às imagens destinadas ao treinamento, enquanto os subconjuntos de validação e teste permaneceram compostos exclusivamente por imagens originais, evitando que versões aumentadas de uma mesma imagem fossem distribuídas entre diferentes partições. Adicionalmente, os modelos de detecção de objetos utilizaram as mesmas imagens em cada subconjunto, assegurando consistência na composição dos dados empregados na comparação. Para a avaliação experimental em campo, foi utilizado também um segundo corredor, cujas imagens não participaram das etapas de treinamento, validação ou teste. Entretanto, os registros disponíveis não permitem confirmar a adoção de uma estratégia específica de agrupamento por sequência de captura, sessão ou objeto físico durante a divisão do conjunto original, permanecendo essa possibilidade como uma limitação metodológica do estudo.

Os experimentos foram desenvolvidos utilizando as implementações disponibilizadas pelas bibliotecas *Ultralytics*, *PyTorch*/*Torchvision* e *Hugging Face Transformers*, conforme a arquitetura avaliada. Na ausência de configurações explicitamente modificadas nos notebooks, consideraram-se as definições padrão dos respectivos frameworks. Para os modelos YOLO11n e YOLO26n, foram identificadas configurações de treinamento com 50 épocas e resolução de entrada de 640 × 640 pixels como apresentado na Tabela 2. Ressalta-se que as configurações padrão podem variar entre versões das bibliotecas e que nem todas as arquiteturas possuem um procedimento de treinamento predefinido.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Tabela 2. Resultados quantitativos reportados para os modelos de detecção de objetos no conjunto independente de teste (holdout test set).

Arquitetura	Épocas	Entrada	Batch size	Otimizador	Learning rate
YOLO11n	100	640 × 640	16	Automático	Automático
YOLO26n	100	640 × 640	16	Automático	Automático
RT-DETR-L (Ultralytics)	100	640 × 640	16	Automático	Automático
SSDLite320 MobileNetV3-Large (Torchvision)	desconhecido	320 × 320	Não definido	Não definido	Não definido
SegFormer MiT-B0 (Hugging Face Trainer)	3	320 × 320	8 por dispositivo	AdamW	0,00005
U-Net (PyTorch)	desconhecido	640 × 640	Não definido	Não definido	Não definido

Fonte: autores

Para a inicialização das arquiteturas de detecção de objetos, foi empregada a estratégia de aprendizagem por transferência (*transfer learning*), utilizando pesos pré-treinados disponibilizados pelos respectivos frameworks. Os modelos YOLO11n e YOLO26n foram inicializados, respectivamente, a partir dos arquivos `yolo11n.pt` e `yolo26n.pt`, enquanto o RT-DETR-L utilizou o arquivo `rtdetr-l.pt`, disponibilizados pela biblioteca *Ultralytics*. Esses modelos partiram de representações visuais previamente aprendidas, associadas ao pré-treinamento no conjunto de dados COCO, sendo posteriormente ajustados (*fine-tuning*) para as cinco classes de objetos definidas no conjunto de dados personalizado de corredores escolares. Para as demais arquiteturas, a identificação completa dos pesos iniciais e das condições de inicialização depende da confirmação dos registros específicos de treinamento.

Avaliação e seleção dos modelos

A avaliação das arquiteturas foi realizada considerando tanto o desempenho preditivo quanto parâmetros associados ao custo computacional. Para os modelos de detecção foram consideradas as métricas mean *Average Precision* com IoU de 0,50 (mAP@50), precisão, recall e F1-score. Para os modelos de segmentação foi

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

adotado o *mean Intersection* over Union (mIoU). O F1-score foi calculado a partir dos valores de precisão e recall. A avaliação quantitativa foi organizada separadamente para as tarefas de detecção de objetos e segmentação semântica, considerando as diferenças entre suas representações de saída e métricas de desempenho. Os modelos YOLO11n, YOLO26, RT-DETR e SSDLite320 MobileNetV3-Large foram avaliados quanto à identificação e localização de objetos por meio de caixas delimitadoras (*bounding boxes*), enquanto *SegFormer* e U-Net foram avaliados quanto à classificação espacial dos pixels por meio de máscaras de segmentação. Dessa forma, os valores de mAP@50 e mIoU foram utilizados exclusivamente nas respectivas tarefas, não sendo considerados diretamente equivalentes para fins de comparação de desempenho preditivo.

As métricas quantitativas de desempenho preditivo foram calculadas, conforme os registros informados pelas responsáveis pelos experimentos, utilizando exclusivamente o subconjunto independente de teste (*holdout* test *set*), correspondente a 10% das imagens originais. Esse subconjunto foi mantido separado durante o treinamento, a validação e o ajuste dos hiperparâmetros, sendo utilizado para a avaliação final dos modelos. Dessa forma, os valores de mAP@50 apresentados para as arquiteturas de detecção de objetos correspondem ao desempenho reportado sobre imagens não utilizadas no processo de treinamento. Os resultados de acompanhamento da aprendizagem durante as épocas de treinamento, apresentados nas curvas de desempenho, constituem uma análise distinta da avaliação final no conjunto de teste.

Também foram considerados GFLOPs, tamanho do modelo, latência de processamento e taxa de quadros por segundo (FPS). A latência por imagem foi determinada considerando os tempos de pré-processamento, inferência e pós-processamento, enquanto o FPS foi obtido a partir do tempo total de processamento de cada quadro.

A complexidade computacional das arquiteturas foi caracterizada por meio do número de parâmetros e da quantidade estimada de operações de ponto flutuante, expressa em GFLOPs. Esses indicadores foram considerados para analisar os requisitos computacionais dos modelos, especialmente em relação à possibilidade

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

de utilização em dispositivos embarcados. Os valores de GFLOPs foram obtidos dos registros experimentais disponíveis, enquanto o número de parâmetros pôde ser recuperado apenas para a arquitetura YOLO11n e são apresentados nas Tabelas 3 e 4. Ressalta-se que os valores de GFLOPs dependem da configuração e da resolução de entrada adotadas em cada arquitetura, não representando, isoladamente, o desempenho efetivo de inferência no dispositivo embarcado.

A partir dessa avaliação preliminar, YOLO11n e YOLO26 foram selecionados para a etapa seguinte dos experimentos por apresentarem os melhores resultados entre as arquiteturas avaliadas. Os dois modelos foram então submetidos aos testes de detecção nos ambientes escolares, permitindo comparar seu comportamento diante das condições consideradas na aplicação proposta.

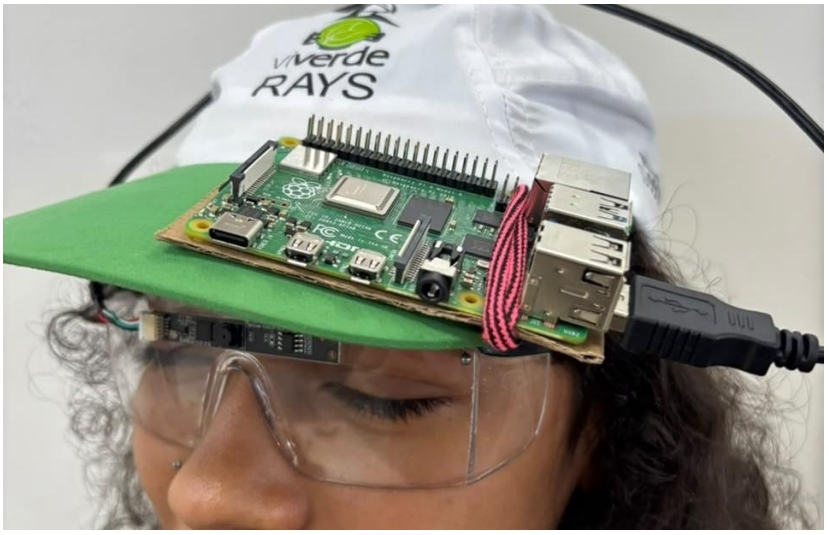
Desenvolvimento do protótipo embarcado

Para o desenvolvimento do protótipo embarcado, foi utilizado um microcomputador *Raspberry* Pi 4 Model B, equipado com processador Broadcom BCM2711, composto por quatro núcleos ARM Cortex-A72 de 64 bits, com frequência de operação de 1,5 GHz, e 4 GB de memória RAM LPDDR4. O dispositivo executou o sistema operacional *Raspberry* Pi OS de 64 bits, baseado na distribuição Debian *Bookworm*. A aquisição das imagens foi realizada por meio de uma câmera USB equipada com sensor CMOS *OmniVision* OV7675, com resolução nativa de 640 × 480 pixels (0,3 megapixel), campo de visão de 66° e ajuste manual de foco (Figura 1). A transmissão dos quadros ocorreu no formato YUY2, enquanto a resolução de entrada utilizada pelos modelos de detecção foi estabelecida em 640 × 640 pixels. A aplicação foi implementada em Python, utilizando a biblioteca *OpenCV* para aquisição e processamento das imagens e a biblioteca *Ultralytics* para execução dos modelos de detecção. O sistema também incorporou uma interface de áudio destinada à emissão de alertas sobre os objetos identificados no ambiente.

Fig.1 - Protótipo desenvolvido.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32



Fonte: autores

A implementação do sistema embarcado foi desenvolvida em Python 3, utilizando o ambiente de desenvolvimento integrado *Thonny* IDE, executado no sistema operacional *Raspberry* Pi OS de 64 bits. Os modelos de detecção foram carregados a partir de arquivos de pesos no formato *PyTorch* (.pt), empregando a biblioteca *Ultralytics* como interface para execução das inferências. A aquisição e o processamento inicial dos quadros foram realizados por meio da biblioteca *OpenCV*, enquanto a etapa de detecção foi conduzida a partir dos modelos previamente treinados e ajustados para as cinco classes de objetos consideradas no estudo. Essa configuração caracteriza a utilização de modelos em formato nativo *PyTorch*, sem documentação de conversão para formatos alternativos de implantação, como ONNX ou *TensorFlow Lite*.

O processamento foi organizado em um fluxo sequencial composto pela captura do quadro da câmera, pré-processamento da imagem, inferência pelo modelo e processamento das detecções. Para cada objeto detectado foram obtidos a classe atribuída, as coordenadas da *bounding box* e o nível de confiança da predição. Essas informações foram utilizadas pela aplicação como entrada para o mecanismo responsável pela geração do alerta sonoro.

O mecanismo de alerta sonoro foi implementado como interface de comunicação entre o sistema de visão computacional e o usuário, utilizando uma lógica determinística de acionamento baseada nas detecções realizadas pelo

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

modelo. Quando um objeto pertencente às classes consideradas no estudo é identificado em um quadro com confiança igual ou superior a 0,50, a aplicação aciona, por meio de chamadas de sistema realizadas em Python, o recurso de reprodução ou síntese de áudio disponibilizado pelo *Raspberry* Pi OS. O alerta consiste em uma mensagem verbal contendo a classe identificada, no formato "[nome do objeto] detectado", como, por exemplo, "Cone detectado" ou "Extintor detectado". Para reduzir a repetição excessiva das mensagens durante o processamento contínuo dos quadros, foi implementado um mecanismo de controle temporal (*debounce*), que estabelece um intervalo mínimo de quatro segundos entre alertas referentes ao mesmo objeto. Dessa forma, o mecanismo fornece informações auditivas sobre a presença de objetos reconhecidos pelo modelo, sem que isso implique, necessariamente, a identificação de uma situação de risco ou a determinação da distância entre o objeto e o usuário.

Avaliação experimental nos ambientes escolares

A avaliação experimental dos modelos YOLO11n e YOLO26 foi realizada considerando dois cenários distintos. O primeiro correspondeu ao corredor utilizado durante a aquisição das imagens que compuseram o conjunto de dados, denominado ambiente conhecido. O segundo correspondeu a outro corredor da instituição, denominado ambiente não observado, que não participou de nenhuma etapa da construção do conjunto de dados, incluindo treinamento, validação e teste dos modelos. Esse segundo ambiente foi utilizado exclusivamente na etapa de avaliação experimental, permitindo investigar o comportamento dos modelos diante de condições ambientais não representadas no conjunto de dados original.

A avaliação experimental foi conduzida por meio de testes repetidos nos dois corredores escolares. Para o modelo YOLO11n, foram realizados seis testes no ambiente conhecido e sete no ambiente não observado, totalizando 13 testes. Para o modelo YOLO26n, foram realizados cinco testes em cada ambiente, totalizando dez testes. Dessa forma, o protocolo experimental compreendeu 23 testes distribuídos entre os dois modelos e os dois ambientes. Em cada teste, o participante percorreu o corredor portando o protótipo, realizando aproximadamente três

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

passagens no sentido de aproximação dos objetos e três passagens no sentido de retorno. Durante os deslocamentos, a distância entre a câmera e os objetos variou de aproximadamente 5 m até a passagem pelo objeto, permitindo observar o comportamento das detecções durante a aproximação. Os experimentos ocorreram sob iluminação artificial, em ambientes adequadamente iluminados, sem regiões de penumbra ou sombras acentuadas relatadas pelas responsáveis pelos testes. As passagens realizadas dentro de cada teste foram consideradas observações repetidas do mesmo procedimento experimental, não constituindo testes independentes adicionais.

Nos dois cenários foram avaliadas as cinco classes definidas para o conjunto de dados. Durante os experimentos, o usuário transitou em postura ereta portando o protótipo, enquanto os objetos eram apresentados à câmera em diferentes condições de distância. Para cada detecção foram registrados a classe identificada e o nível de confiança atribuído pelo modelo. Também foram observadas as situações em que uma das classes avaliadas não foi reconhecida.

A avaliação experimental considerou variações na distância entre a câmera e os objetos, compreendendo um intervalo operacional de 1,0 m a 5,0 m, conforme as condições estabelecidas para os testes nos ambientes escolares. Essa variação permitiu observar alterações nos níveis de confiança atribuídos pelos modelos às detecções realizadas. Os resultados obtidos nos dois corredores foram posteriormente comparados quanto às classes reconhecidas, aos níveis de confiança das predições e ao comportamento dos modelos diante da mudança de cenário. Entretanto, os registros experimentais disponíveis não permitem associar sistematicamente cada resultado a uma distância específica dentro do intervalo avaliado, limitando a análise quantitativa da influência desse fator.

Para a etapa de inferência em tempo real dos modelos YOLO11n e YOLO26n, foi adotado um limiar mínimo de confiança (*confidence threshold*) de 0,50 (50%), sendo consideradas válidas apenas as predições com confiança igual ou superior a esse valor. O processamento das detecções considerou um limiar de interseção sobre união (*Intersection over Union* – IoU) de 0,50 para o procedimento de supressão de detecções redundantes (*Non-Maximum Suppression* – NMS),

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

conforme a configuração informada para os experimentos. As imagens foram processadas com resolução de entrada de 640 × 640 pixels. Esses parâmetros foram utilizados como referência para a padronização das condições de inferência e para a análise das detecções realizadas nos dois corredores escolares.

3. Resultados e discussão

Treinamento e avaliação das arquiteturas

Inicialmente, foram investigadas seis arquiteturas de visão computacional, organizadas conforme a tarefa desempenhada. Para a detecção de objetos, foram avaliados YOLO11nn, YOLO26n, RT-DETR-L e SSDLite320 MobileNetV3-Large, cujos resultados quantitativos são apresentados na Tabela 3. Para a segmentação semântica, foram avaliados *SegFormer* e U-Net, com resultados apresentados na Tabela 4. Essa organização permite analisar separadamente o desempenho preditivo das arquiteturas, considerando as métricas específicas de cada tarefa.

A avaliação quantitativa dos modelos de detecção considerou a métrica mean *Average Precision* (mAP), com destaque para o mAP@50, calculado com limiar de interseção sobre união (IoU) de 0,50. Complementarmente, o mAP@50-95 representa a média dos valores de precisão média obtidos em diferentes limiares de IoU, de 0,50 a 0,95, constituindo uma avaliação mais rigorosa da qualidade de localização dos objetos. Os valores de mAP@50 apresentados na Tabela 3 correspondem aos resultados reportados para o conjunto independente de teste. Entretanto, os registros experimentais disponíveis não permitiram recuperar os valores numéricos de mAP@50-95 para todas as arquiteturas nesse mesmo conjunto. Por esse motivo, a comparação quantitativa principal foi mantida com base no mAP@50, enquanto os valores de mAP@50-95 apresentados nas curvas de treinamento foram considerados exclusivamente para análise do comportamento dos modelos durante a aprendizagem, sem equivalência direta com os resultados da avaliação final.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Tabela 3 — Resultados relatados para os modelos de detecção de objetos

Modelo	mAP@50	Precision	Recall	F1-score	GFLOPs	Latência (ms)	FPS	Peso (MB)
YOLO11nn	0,995	0,946	1,000	0,9562	6,3	8,7	94,3	5,5
YOLO26n	1,000	0,900	1,000	0,9330	5,2	7,4	112,4	4,8
RT-DETR-L	0,995	0,953	1,000	0,9762	103,5	23,4	41,3	66,2
SSDLite320 MobileNetV3 -Large	0,233	0,310	0,230	0,2627	0,62	33,43	29,91	13,42

Fonte: autores

A avaliação dos modelos de detecção também considerou as métricas de precisão (*precision*), revocação (*recall*) e F1-score, permitindo analisar diferentes aspectos do desempenho preditivo (Tabela 3). A precisão representa a proporção de detecções positivas corretas entre aquelas realizadas pelo modelo, enquanto a revocação expressa a proporção de objetos efetivamente identificados entre os existentes no conjunto avaliado. O F1-score, por sua vez, combina precisão e revocação por meio da média harmônica, oferecendo uma medida complementar de equilíbrio entre falsos positivos e falsos negativos. No contexto da aplicação assistiva investigada, a revocação apresenta particular relevância, pois a não identificação de um objeto pode comprometer a capacidade do sistema de emitir um alerta ao usuário.

Tabela 4 — Resultados relatados para os modelos de segmentação semântica

Modelo	mIoU	F1-score	GFLOPs	Latência (ms)	FPS	Peso (MB)
SegFormer	0,9077	0,9516	8,4	12,5	80,0	14,3
U-Net	0,8776	0,9348	16,5	12,5	54,9	90,0

Fonte: autores

A análise das arquiteturas considerou os resultados experimentais preservados durante o desenvolvimento do projeto e os procedimentos de treinamento documentados nos respectivos notebooks. Embora tenham sido recuperadas informações referentes às seis arquiteturas avaliadas, nem todos os

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

arquivos originais de execução, incluindo históricos completos de treinamento e resultados individuais de validação e teste, permaneceram disponíveis. Dessa forma, os resultados cuja rastreabilidade experimental não pôde ser integralmente estabelecida foram considerados apenas como evidências preliminares, não sendo utilizados isoladamente para sustentar conclusões definitivas sobre a superioridade de uma arquitetura.

Entre os modelos de detecção, YOLO11nn, YOLO26n e RT-DETR-L apresentaram valores elevados de mAP@50, enquanto SSDLite320 MobileNetV3-Large apresentou resultado inferior. O RT-DETR-L também apresentou maior complexidade computacional, expressa em GFLOPs e tamanho do modelo. Entre os segmentadores, *SegFormer* apresentou valores superiores de mIoU e F1-score em relação à U-Net, além de menor complexidade computacional nos indicadores apresentados.

Considerando que o objetivo do protótipo assistivo consiste na identificação e localização de objetos por meio de caixas delimitadoras, a etapa experimental em corredores escolares foi restrita às arquiteturas de detecção. Nesse contexto, YOLO11nn e YOLO26n foram selecionados por apresentarem resultados preditivos elevados e menor complexidade computacional em relação ao RT-DETR-L, permitindo investigar o comportamento de duas arquiteturas da família YOLO em ambientes conhecidos e não observados. Os resultados de segmentação foram mantidos como parte da investigação preliminar, sem comparação direta de suas métricas preditivas com aquelas utilizadas para detecção de objetos.

Caracterização dos cenários de avaliação

Após a etapa de treinamento e avaliação das arquiteturas, os modelos YOLO11n e YOLO26 foram selecionados para os experimentos de detecção nos corredores escolares. A avaliação considerou as cinco classes definidas para o conjunto de dados: *cleaning cart* (carrinho de limpeza), *fire extinguisher* (extintor), *safety cone* (cone de sinalização), *water fountain* (bebedouro) e *trash can* (lixeira).

Os experimentos foram conduzidos em dois cenários. O primeiro correspondeu ao corredor utilizado durante a etapa de aquisição das imagens que

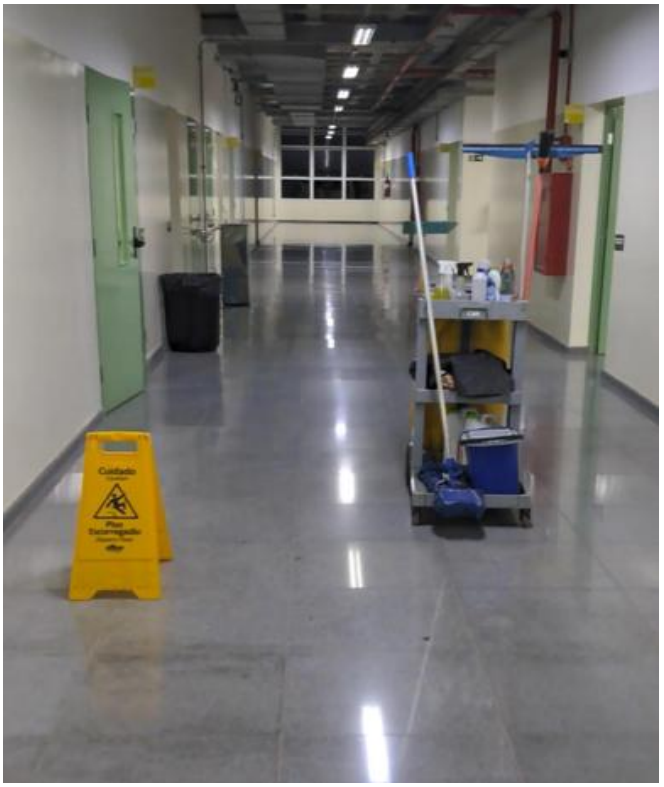
REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

compuseram o conjunto de dados, sendo denominado neste trabalho ambiente conhecido (Figura 2). O segundo correspondeu a outro corredor da instituição, cujas características e objetos não haviam sido utilizados durante a coleta das imagens de treinamento, sendo denominado ambiente não observado (Figura 3). Essa configuração permitiu avaliar o comportamento dos modelos tanto em condições semelhantes às representadas no conjunto de dados quanto diante de uma mudança de ambiente.

Durante os experimentos, foram consideradas as detecções realizadas com o usuário transitando em postura ereta e em condições adequadas de proximidade dos objetos avaliados. Para cada detecção, foram registrados a classe identificada e o nível de confiança atribuído pelo modelo.

Fig. 2 – Ambiente conhecido.

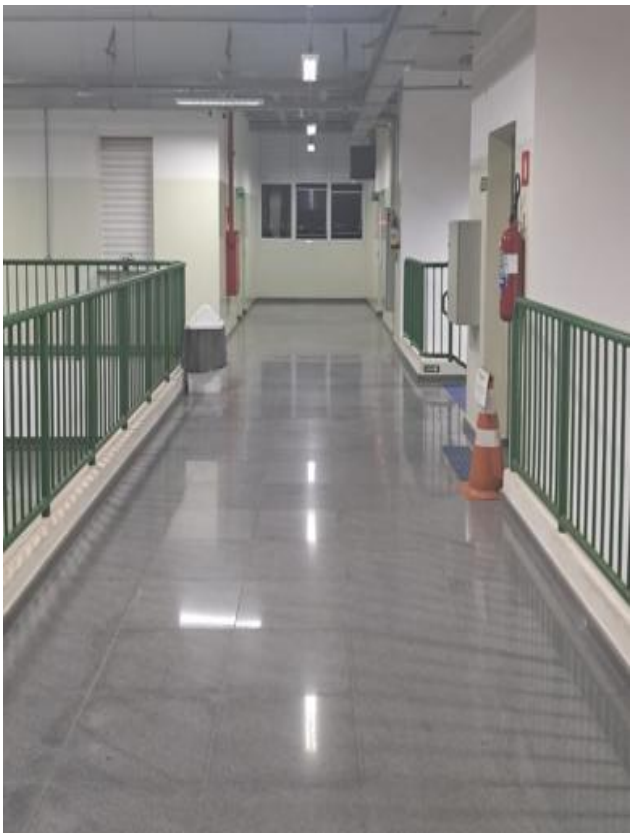


Fonte: autores.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Fig. 3 – Ambiente não observado.



Fonte: autores.

Desempenho dos modelos durante o treinamento

A evolução das métricas de treinamento dos modelos selecionados é apresentada nas Figuras 4 e 5. Tanto o YOLO11n quanto o YOLO26 apresentaram crescimento acentuado do *mAP* nas primeiras épocas, atingindo posteriormente valores próximos de 100%. Esse comportamento indica que ambos os modelos conseguiram aprender as características das classes presentes no conjunto de dados utilizado no treinamento.

Para o YOLO26, o mAP@50-95 apresentou crescimento progressivo e posterior estabilização entre aproximadamente 70% e 75%. No YOLO11n, essa métrica estabilizou-se em valores entre aproximadamente 65% e 70%. Considerando essa métrica, portanto, o YOLO26 apresentou desempenho ligeiramente superior durante o treinamento.

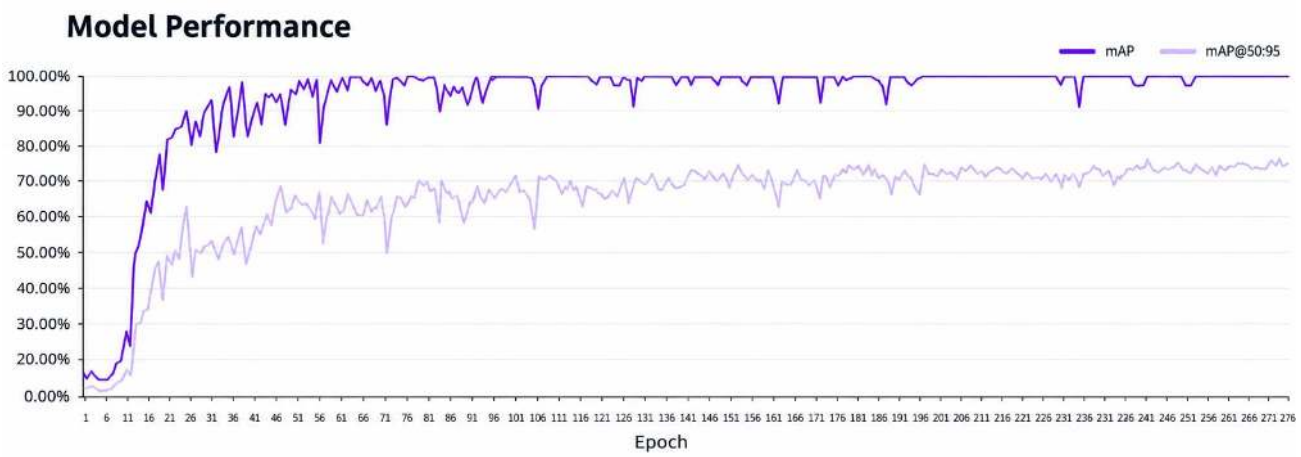
A evolução da *class loss* também apresentou comportamento semelhante nos

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

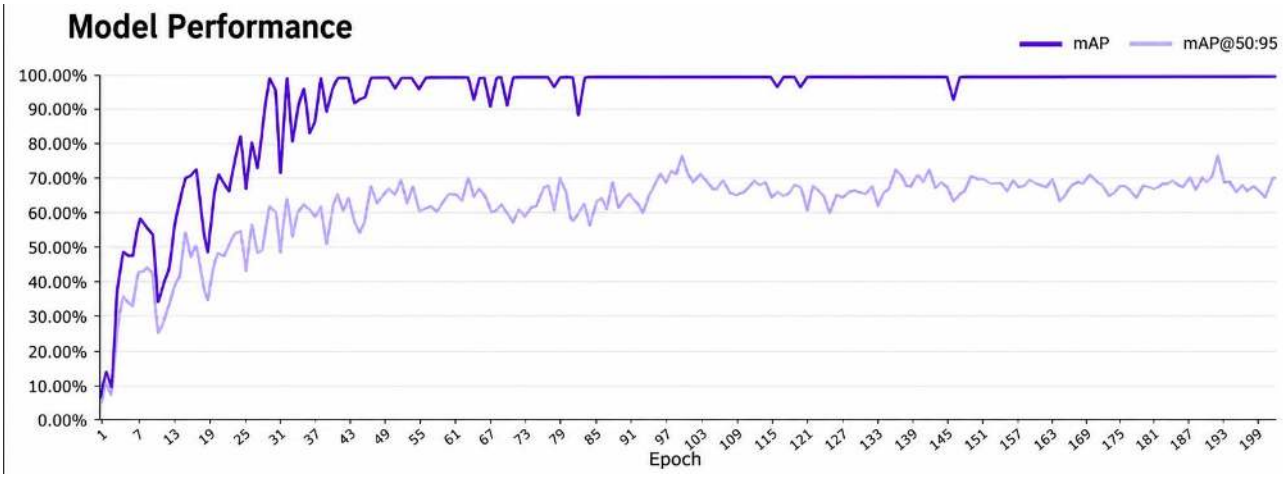
dois modelos. Observou-se uma redução acentuada nas primeiras épocas, seguida de estabilização em valores baixos ao longo do restante do treinamento. Em conjunto com a evolução do *mAP*, esse comportamento indica a convergência dos modelos durante o processo de treinamento. Entretanto, os resultados obtidos nessa etapa devem ser analisados em conjunto com os experimentos realizados nos corredores, uma vez que o desempenho sobre o conjunto de dados não necessariamente representa o comportamento do modelo diante de imagens obtidas em outros ambientes.

Fig. 4 – Desempenho do Yolo26.



Fonte: autores.

Fig. 5 – Desempenho do YOLO11n.



Fonte: autores.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Desempenho da detecção nos ambientes avaliados

A Tabela 5 apresenta os níveis de confiança obtidos pelos modelos YOLO11n e YOLO26 para as cinco classes nos ambientes conhecido e não observado.

Tabela 5 - Níveis de confiança das detecções obtidas pelos modelos YOLO11n e YOLO26 nos dois ambientes avaliados.

Objeto	YOLO26 Ambiente conhecido	YOLO11n Ambiente conhecido	YOLO26 Ambiente não observado	YOLO11n Ambiente não observado
Cone de segurança	94%	51%	67%	87%
Extintor de incêndio	88%	83%	66%	67%
Carrinho de limpeza	86%	74%	77%	72%
Lixeira	Não reconhecida	62%	Não reconhecida	Não reconhecida
Bebedouro	Não reconhecido	62%	Não reconhecido	Não reconhecido

Fonte: autores

No ambiente conhecido, o YOLO26 detectou três das cinco classes avaliadas. O cone de sinalização apresentou confiança de 94%, seguido pelo extintor, com 88%, e pelo carrinho de limpeza, com 86%. A lixeira e o bebedouro não foram reconhecidos nesse cenário. O YOLO11n, por sua vez, detectou as cinco classes, apresentando 83% de confiança para o extintor, 74% para o carrinho de limpeza, 62% para a lixeira, 62% para o bebedouro e 51% para o cone de sinalização.

Os resultados mostram, portanto, comportamentos distintos entre os modelos no ambiente conhecido. Entre as classes reconhecidas por ambos, o YOLO26 apresentou níveis de confiança superiores para cone, extintor e carrinho de limpeza. Entretanto, somente o YOLO11n foi capaz de detectar todas as cinco classes avaliadas nesse ambiente.

No ambiente não observado durante a composição do conjunto de dados, ambos os modelos detectaram três das cinco classes. O YOLO26 apresentou confiança de 77% para o carrinho de limpeza, 67% para o cone de sinalização e 66% para o extintor. Para o YOLO11n, foram obtidos 87% para o cone de sinalização, 72% para o carrinho de limpeza e 67% para o extintor. Lixeira e bebedouro não foram detectados por nenhum dos modelos nesse cenário.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Durante a avaliação experimental nos corredores escolares, foram observadas alterações nos níveis de confiança das detecções em função da aproximação da câmera em relação aos objetos. Os deslocamentos foram iniciados a aproximadamente 5 m dos objetos, com aproximação progressiva durante as passagens. Para o YOLO11n, foram relatadas detecções a distâncias próximas de 3 m, com níveis de confiança superiores a 66% em determinadas observações no ambiente conhecido, e tendência de aumento desses valores com a redução da distância. No ambiente não observado, foi relatada uma redução aproximada dos níveis de confiança, mantendo-se a tendência de aumento durante a aproximação. Para o YOLO26n, as responsáveis relataram detecções a distâncias próximas de 2 m, com níveis de confiança iguais ou superiores a 80% em determinadas observações no ambiente conhecido, além de redução dos valores de confiança no ambiente não observado.

A avaliação experimental realizada nos corredores permitiu observar a identificação e a ausência de reconhecimento de determinadas classes de objetos pelos modelos YOLO11n e YOLO26n. Entretanto, os registros preservados não contemplam a contabilização sistemática de verdadeiros positivos (TP), falsos positivos (FP) e falsos negativos (FN) para todas as observações realizadas em campo. Dessa forma, não foi possível calcular métricas agregadas de precisão, revocação e F1-score específicas dessa etapa experimental. Os resultados apresentados devem, portanto, ser interpretados como evidências exploratórias do comportamento dos modelos nos ambientes avaliados, não constituindo uma avaliação quantitativa completa de sua capacidade de detecção em condições operacionais reais.

Comparação entre YOLO11n e YOLO26

A comparação dos resultados evidencia que os modelos apresentaram características distintas. No ambiente conhecido, o *YOLO26* produziu os maiores níveis de confiança para as três classes detectadas por ambos os modelos. A diferença foi particularmente expressiva para o cone de sinalização, cuja confiança foi de 94% no *YOLO26* e 51% no *YOLO11n*. Para o extintor, foram obtidos 88% e

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

83%, respectivamente, enquanto para o carrinho de limpeza os valores foram de 86% e 74%.

Entretanto, níveis de confiança mais elevados nas classes detectadas não implicaram maior cobertura das classes avaliadas. Enquanto o *YOLO26* reconheceu três das cinco classes no ambiente conhecido, o *YOLO11n* detectou as cinco. Esse resultado é relevante para a aplicação proposta, uma vez que a ausência de detecção de um objeto potencialmente relevante pode ser tão importante quanto o nível de confiança atribuído às detecções realizadas.

Ressalta-se que os níveis de confiança (*confidence scores*) apresentados nos experimentos representam pontuações atribuídas pelos modelos a detecções individuais e não devem ser interpretados como medidas de acurácia, precisão ou desempenho global das arquiteturas. Além disso, essas pontuações não são necessariamente calibradas entre diferentes modelos, de modo que valores mais elevados não implicam, isoladamente, maior capacidade de detecção ou melhor generalização. Assim, a comparação entre YOLO11n e YOLO26n deve considerar conjuntamente as métricas quantitativas obtidas no conjunto independente de teste, a identificação das classes nos ambientes avaliados e a ocorrência de falhas de reconhecimento. Nesse contexto, os níveis de confiança são utilizados apenas como indicadores complementares do comportamento das predições, e não como critério principal para estabelecer a superioridade de uma arquitetura.

A mudança para o ambiente não observado modificou esse comportamento. Nesse cenário, ambos os modelos reconheceram três classes. O *YOLO11n* apresentou maior confiança para o cone de sinalização (87% contra 67%) e valores próximos aos do *YOLO26* para o extintor (67% contra 66%). Para o carrinho de limpeza, o *YOLO26* apresentou confiança de 77%, enquanto o YOLO11n atingiu 72%.

Destaca-se ainda que lixeira e bebedouro não foram reconhecidos por nenhum dos modelos no ambiente não observado, apesar de terem sido detectados pelo *YOLO11n* no ambiente conhecido. As diferenças visuais entre os objetos presentes nos dois ambientes constituem uma possível explicação para esse comportamento; contudo, os experimentos realizados não permitem estabelecer

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

isoladamente a causa dessas falhas de detecção. Os resultados indicam, portanto, uma redução da capacidade de reconhecimento de determinadas classes quando os modelos são submetidos a condições diferentes daquelas representadas no conjunto de dados.

De forma geral, o *YOLO26* apresentou maiores níveis de confiança para algumas classes, especialmente no ambiente conhecido, enquanto o *YOLO11n* apresentou maior cobertura das classes nesse mesmo cenário. Os resultados no ambiente não observado mostram ainda que a análise baseada exclusivamente nas métricas obtidas durante o treinamento não é suficiente para caracterizar o comportamento dos modelos em condições distintas das utilizadas na construção do conjunto de dados.

Implicações para o protótipo embarcado

Os resultados obtidos apresentam implicações diretas para a utilização dos modelos no protótipo assistivo. Em uma aplicação destinada à identificação de objetos potencialmente relevantes para a navegação, não apenas o nível de confiança das detecções deve ser considerado, mas também a capacidade do sistema de reconhecer diferentes classes diante de variações do ambiente e da distância entre a câmera e os objetos.

Nesse sentido, embora o *YOLO26* tenha apresentado valores superiores de mAP@50-95 durante o treinamento e maiores níveis de confiança para determinadas classes, o *YOLO11n* apresentou a vantagem de reconhecer todas as cinco classes no ambiente conhecido. Por outro lado, a ausência de detecção da lixeira e do bebedouro por ambos os modelos no ambiente não observado evidencia uma limitação relevante para a aplicação proposta e indica a necessidade de ampliar a diversidade de imagens utilizadas no treinamento, particularmente quanto às variações visuais dos objetos e dos ambientes nos quais o sistema deverá operar.

Neste estudo, a identificação de objetos por meio dos modelos de visão computacional não deve ser interpretada como uma avaliação automática de situações de perigo. As arquiteturas investigadas foram treinadas para reconhecer cinco classes de objetos potencialmente relevantes para a navegação em corredores

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

escolares, sem incorporar mecanismos específicos para determinar sua posição em relação à trajetória do usuário, estimar a distância absoluta ou identificar zonas de colisão. Assim, a presença de um objeto no campo de visão da câmera não implica, necessariamente, que ele represente um obstáculo à circulação. A determinação efetiva de risco exigiria informações espaciais e contextuais adicionais, não contempladas no escopo experimental deste trabalho.

Considerando a finalidade assistiva do protótipo, a análise dos resultados deve atribuir especial importância à revocação (recall) e à ocorrência de falsos negativos, uma vez que a ausência de reconhecimento de um objeto pode impedir a emissão de um alerta necessário ao usuário. Nesse contexto, a taxa de falsos negativos, definida pela razão entre os objetos não detectados e o total de objetos efetivamente presentes, constitui um indicador particularmente relevante para avaliar as limitações do sistema. Embora os resultados quantitativos obtidos no conjunto independente de teste permitam caracterizar o desempenho preditivo das arquiteturas, as falhas de reconhecimento observadas nos corredores evidenciam a necessidade de avaliar essa capacidade também em condições operacionais distintas das utilizadas no desenvolvimento dos modelos. Destaca-se, entretanto, que a ausência de registros completos de verdadeiros positivos e falsos negativos nos experimentos em campo impossibilita calcular uma taxa de falsos negativos específica para essa etapa.

Os experimentos também demonstraram que a distância entre a câmera e os objetos influencia os níveis de confiança das detecções, aspecto particularmente relevante em um dispositivo vestível, no qual a posição da câmera varia continuamente durante o deslocamento do usuário. Dessa forma, os resultados obtidos indicam o potencial da utilização dos modelos para a identificação das classes consideradas, mas também evidenciam limitações de generalização que devem ser consideradas na evolução do protótipo e na ampliação do conjunto de dados.

Embora a avaliação experimental tenha contemplado distâncias entre 1,0 m e 5,0 m da câmera aos objetos, essa condição corresponde ao posicionamento adotado durante os testes e não à capacidade do protótipo de estimar

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

automaticamente a distância dos elementos identificados. Os modelos de detecção empregados fornecem informações sobre a classe reconhecida, as coordenadas da caixa delimitadora (*bounding box*) e o nível de confiança da predição, mas não realizam estimativa de profundidade ou determinação da distância absoluta entre a câmera e os objetos. Consequentemente, o sistema não dispõe de informações espaciais suficientes para determinar a proximidade efetiva de um obstáculo ou avaliar automaticamente o risco de colisão durante o deslocamento do usuário. Como possibilidade de aprimoramento, trabalhos futuros poderão investigar a integração de técnicas de estimativa monocular de profundidade, câmeras estereoscópicas ou sensores específicos de distância, como ultrassônicos e LiDAR, permitindo associar a identificação visual dos objetos à sua localização espacial.

A análise da viabilidade computacional do protótipo deve considerar as limitações de processamento da plataforma embarcada. Embora os resultados preliminares apresentados na Tabela 3 indiquem latências de 8,7 ms para o YOLO11n e 7,4 ms para o YOLO26n, esses valores não foram identificados como medições realizadas diretamente no *Raspberry* Pi 4 utilizado no protótipo. Estudos independentes demonstram que o desempenho de arquiteturas YOLO nessa plataforma pode variar substancialmente conforme o ambiente de execução, o formato de implementação e os recursos computacionais disponíveis. Safonova et al. (2026), ao avaliarem diferentes arquiteturas YOLO em um *Raspberry* Pi 4B, registraram latências de 374,24 ms e 483,75 ms para o YOLO11n, utilizando, respectivamente, os formatos *OpenVINO* e NCNN. Em uma aplicação de visão computacional assistiva destinada a pessoas com deficiência visual, Noor et al. (2025) avaliaram o YOLOv11 em um *Raspberry* Pi 4 Model B, obtendo uma taxa média de processamento de 1,87 FPS com um modelo ONNX quantizado e execução em um único núcleo de CPU, equivalente a aproximadamente 535 ms por quadro. Esses resultados evidenciam que o desempenho computacional depende não apenas da arquitetura selecionada, mas também das condições de implantação e execução. Entretanto, como os protocolos experimentais não são diretamente equivalentes, os resultados externos não permitem estimar a latência efetiva do protótipo desenvolvido. Dessa forma, a avaliação da capacidade de resposta em

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

tempo real permanece condicionada à realização de medições específicas na plataforma embarcada utilizada neste trabalho.

Por fim, a taxa de processamento de quadros por segundo (*frames* per *second* – FPS) constitui um indicador relevante para avaliar a capacidade de atualização das detecções em aplicações assistivas. Embora a Tabela 3 apresente valores de 94,3 FPS para o YOLO11n e 112,4 FPS para o YOLO26n, esses resultados não foram identificados como medições realizadas diretamente no *Raspberry* Pi 4 utilizado no protótipo. A literatura demonstra que o desempenho de arquiteturas YOLO em plataformas embarcadas depende das condições de execução e das estratégias de otimização adotadas. Safonova et al. (2026) relataram latências de 374,24 ms e 483,75 ms para o YOLO11n em um *Raspberry* Pi 4B, utilizando, respectivamente, *OpenVINO* e NCNN, correspondentes a taxas teóricas aproximadas de 2,67 e 2,07 quadros por segundo, considerando exclusivamente os tempos de processamento informados. Em uma aplicação de detecção de objetos voltada a pessoas com deficiência visual, Noor et al. (2025) registraram uma taxa de 1,87 FPS para uma configuração do YOLOv11 executada em um Raspberry Pi 4 Model B. Esses resultados evidenciam a necessidade de distinguir o desempenho preliminar dos modelos da capacidade efetiva de processamento do sistema embarcado. Entretanto, diferenças entre arquiteturas, configurações e protocolos experimentais impedem comparações diretas. Assim, a ausência de medições sistemáticas de FPS no *Raspberry* Pi utilizado neste estudo limita a caracterização quantitativa da capacidade de atualização das detecções em condições reais de operação.

4. Conclusões

Este trabalho apresentou o desenvolvimento e a avaliação de uma solução de visão computacional voltada à identificação de objetos potencialmente relevantes para a navegação em corredores escolares, visando sua aplicação em um protótipo assistivo vestível. A partir da construção de um conjunto de dados específico do domínio e da avaliação de diferentes arquiteturas, os modelos *YOLO11n* e *YOLO26* foram selecionados para os experimentos conduzidos em um ambiente conhecido e em um ambiente não observado nessa etapa.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Os resultados demonstraram comportamentos distintos entre os modelos. O *YOLO26* apresentou valores superiores de mAP@50-95 durante o treinamento e maiores níveis de confiança para determinadas classes no ambiente conhecido. O *YOLO11n*, entretanto, apresentou maior cobertura nesse cenário, sendo capaz de detectar as cinco classes avaliadas. No ambiente não observado, ambos os modelos detectaram três das cinco classes, não reconhecendo lixeira e bebedouro. Também foram observadas variações nos níveis de confiança em função da distância entre a câmera e os objetos. Esses resultados mostram que métricas obtidas durante o treinamento, isoladamente, não são suficientes para caracterizar o comportamento dos modelos nas condições de aplicação.

Dessa forma, a hipótese proposta foi parcialmente sustentada. Os experimentos demonstraram o potencial de modelos treinados com dados específicos do ambiente escolar para a identificação dos objetos considerados, inclusive diante de mudanças no cenário de aplicação. Entretanto, as falhas de detecção observadas no ambiente não observado evidenciam limitações de generalização que ainda precisam ser superadas. Além disso, embora o protótipo tenha sido desenvolvido sobre uma plataforma embarcada, os resultados apresentados não permitem concluir de forma definitiva sobre seu desempenho em tempo real, uma vez que a avaliação experimental não contemplou uma análise sistemática de latência e FPS durante a execução no *Raspberry* Pi.

Cabe destacar que os resultados apresentados devem ser interpretados como evidências de uma prova de conceito de visão computacional embarcada com finalidade assistiva, desenvolvida e avaliada preliminarmente em condições experimentais restritas a cinco classes de objetos e dois ambientes escolares, não constituindo a validação de um sistema completo de auxílio à navegação. Uma das principais limitações deste estudo consiste na ausência de avaliações funcionais sistemáticas conduzidas com pessoas com deficiência visual, público-alvo da solução proposta. Embora os experimentos tenham permitido investigar o comportamento dos modelos na identificação de objetos em corredores escolares, não foram avaliados aspectos diretamente relacionados à experiência desses usuários, como a compreensão dos alertas sonoros, o tempo disponível para reagir

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

às informações fornecidas, a possibilidade de sobrecarga auditiva e a contribuição efetiva do dispositivo para a mobilidade e a segurança durante o deslocamento. Consequentemente, os resultados não permitem demonstrar a efetividade assistiva ou a redução de riscos proporcionada pelo protótipo. A consolidação da proposta como tecnologia assistiva dependerá de estudos futuros com participação do público-alvo, contemplando avaliações de usabilidade, acessibilidade, segurança e desempenho funcional em situações reais de utilização.

Como continuidade do trabalho, propõe-se ampliar o conjunto de dados, incorporando maior diversidade de ambientes, objetos, perspectivas e distâncias, bem como realizar uma avaliação sistemática do desempenho computacional dos modelos diretamente na plataforma embarcada. Também se recomenda ampliar os experimentos com usuários e em diferentes condições de circulação, permitindo avaliar não apenas o desempenho dos modelos de visão computacional, mas a efetividade do sistema assistivo em condições mais próximas de sua utilização real.

Disponibilidade de dados

Os dados utilizados neste projeto estão disponíveis em:
https://github.com/Rafaela-Mello/embedded-assistive-vision

Referências

Alsultan O. K. T. and M. T. Mohammad. **A deep learning-based assistive system for the visually impaired using YOLO-V7**. Rev. Intell. Artif., vol. 37, no. 4, pp. 901–906, 2023, doi: https://doi.org/10.18280/ria.370409.

Balthazar, Glauber da Rocha; Silveira, Robson Mateus Freitas; Silva, Iran José Oliveira da. **Design and Prototyping of a Robotic Structure for Poultry Farming**. Agriengineering, [S.L.], v. 7, n. 7, p. 233, 11 jul. 2025. MDPI AG. http://dx.doi.org/10.3390/agriengineering7070233.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Castro Junior, Sérgio Luís de e Balthazar, Glauber da Rocha e Silva, Iran José Oliveira da. **Diagnóstico preditivo em tempo real do conforto térmico de animais de produção em Sistema Operacional Android**. 2019, Anais. Lavras: DEA-UFLA, 2019. p. 1091 il. Disponível em: http://www.infobibos.com/anais/siapas2/EBOOK_SIAPAS_SIMCRA.pdf. Acesso em: 07 out. 2026.

Gassenn L. et al. **Aplicação de modelos de aprendizado profundo na estimativa de relações espaciais dos objetos para auxiliar pessoas com deficiência visual.** Proc. Simp. Bras. Comput. Aplic. Saúde (SBCAS), 2024.

Klann D., Fernandes A. M. R., Silva E. A., and Parreira W. D. **Explorando algoritmos de visão computacional em tecnologias assistivas: uma revisão sistemática da literatura**. J. Health Inform., 2024.

Kraetzer, T. L., & Balthazar, G. R. (2021). FISHBOARD: **An electronic device for analysis of productive data in pisciculture (fish-farming)**. Brazilian Journal of Development, 7(3), 28513–28533. https://doi.org/10.34117/bjdv7n3-526.

Li X. et al. **A dataset for the visually impaired walk on the road**. Displays, vol. 79, Art. no. 102486, 2023.

Litoriya R. et al. **Implementing visual assistant using YOLO and SSD for visually-impaired persons**. J. Autom. Mob. Robot. Intell. Syst., vol. 17, no. 4, pp. 79–87, 2023.

NOOR, Ayman; ALMUKHALFI, Hanan; SOUZA, Arthur; NOOR, Talal H. **Towards a real-time indoor object detection for visually impaired users using Raspberry Pi 4 and YOLOv11: a feasibility study**. Computer Modeling in Engineering & Sciences, v. 144, n. 3, p. 3085–3111, 2025. DOI: https://doi.org/10.32604/cmes.2025.068393.

REMUNOM
ISSN 2178-6925

Received: 13/09/2026 - **Accepted:** 13/10/2026
Vol: 13.22
DOI: 10.66104/0xbft773
Pages: 1-32

Pickett W. et al. **Multiple risk behavior and injury: an international analysis of young people**. Arch. Pediatr. Adolesc. Med., vol. 156, no. 8, pp. 786–793, 2002.

Safonova, Anastasiia et al. **Deep learning for UAV thermal bird detection: benchmarking YOLOv8–YOLOv26**. Remote Sensing Applications: Society and Environment, v. 42, art. 101981, 2026. DOI: https://doi.org/10.1016/j.rsase.2026.101981.

Santiago M. T. F., Faria M. V. B., Malva M. V. P. r, and Nascimento L. O. **Object identification for visually impaired**. J. Bioeng. Technol. Health, vol. 6, no. 4, pp. 296–300, 2023.

Silva G. L. and Silva L. G. P. **Óculos de assistência na locomoção para deficientes visuais.** Rev. GeTeC, vol. 21, pp. 139–158, 2025.

Stadtmüller S. et al. **Approaching the causes of unintentional injuries in the school environment: a panel analysis of survey data from Germany**. J. Sch. Health, vol. 92, no. 2, pp. 148–156, 2022, doi: https://doi.org/10.1111/josh.13112.

ULTRALYTICS. **Ultralytics YOLO11**. 2024. Disponível em: https://docs.ultralytics.com/models/yolo11. Acesso em: 13 out. 2026.

ULTRALYTICS. **Ultralytics YOLO26**. 2026. Disponível em: https://docs.ultralytics.com/models/yolo26. Acesso em: 13 out. 2026.

Valipoor M. M. and Antonio A. De. **Recent trends in computer vision-driven scene understanding for VI/blind users: a systematic mapping**. Univers. Access Inf. Soc., vol. 22, no. 3, pp. 983–1005, 2023, doi: https://doi.org/10.1007/s10209-022-00868-w.