

16º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2025

DETECÇÃO DO USO DE DISPOSITIVOS MÓVEIS NAS ESCOLAS: UMA ABORDAGEM UTILIZANDO A VISÃO COMPUTACIONAL

Giovana T. dos Reis¹, João F. M. Bernardelli², Luiz G. Sgobi³, Thiago M. Dassena⁴, Márcio A. Teixeira⁵

¹ Discente do curso Técnico em Redes de Computadores Integrado ao Ensino Médio, Campus Catanduva (CTD), giovana.tudes@ifsp.edu.br.

² Discente do curso Técnico em Redes de Computadores Integrado ao Ensino Médio, Campus Catanduva (CTD), bernardelli.joao@aluno.ifsp.edu.br.

³ Discente do curso Técnico em Redes de Computadores Integrado ao Ensino Médio, Campus Catanduva (CTD), luiz.gustavo@aluno.ifsp.edu.br.

⁴ Discente do curso Técnico em Redes de Computadores Integrado ao Ensino Médio, Campus Catanduva (CTD), dassena.thiago@aluno.ifsp.edu.br.

⁵ Docente do IFSP Campus Catanduva (CTD), Eixo Informática, marcio.andrey@ifsp.edu.br.

Área de conhecimento (Tabela CNPq): 1.03.03.04-9 Sistemas de Informação

RESUMO: O uso inadequado de dispositivos móveis no ambiente escolar representa um desafio para o processo de aprendizagem dos alunos, mesmo com legislações que proíbam o seu uso, como a Lei Federal 15.100/2025. Muitas vezes, a fiscalização feita manualmente é ineficaz devido à falta de recursos humanos. Pensando nisso, esse trabalho desenvolveu um sistema automatizado para auxiliar esse monitoramento. A solução utiliza câmeras de segurança integradas a um modelo de Inteligência Artificial (IA) baseado em Visão Computacional (VC), capaz de identificar quando os alunos estão utilizando celulares e gerar ocorrências. Tais ocorrências, gerenciadas por uma API, são apresentadas em um *dashboard* web, facilitando o trabalho dos inspetores. O sistema foi desenvolvido utilizando o modelo RF-DETR para detecção de objetos, Spring Boot para o *back-end*, React/Next.js para o *front-end*, além de contar com o PostgreSQL como base de dados para armazenar os dados da aplicação. Como resultado, o monitoramento automatizado tornou o trabalho de fiscalização dos inspetores mais eficiente, aumentando a eficácia no cumprimento da lei e melhorando o ambiente de aprendizado.

PALAVRAS-CHAVE: visão computacional; monitoramento automatizado; inteligência artificial; detecção de objetos; celular.

DETECTION OF MOBILE DEVICE USAGE IN SCHOOLS: AN APPROACH USING COMPUTER VISION

ABSTRACT: The inappropriate use of mobile devices in the school environment poses a challenge to the students' learning process, even with legislation prohibiting their use, such as Federal Law 15.100/2025. Often, manual supervision proves ineffective due to a lack of human resources. In that context, this work developed an automated system to assist in monitoring. The solution utilizes security cameras integrated with an Artificial Intelligence (AI) model based on Computer Vision (CV), which can identify when students are using mobile phones and generating incident reports. These reports are managed by an API, which displays them in a web dashboard, facilitating the inspectors' work. The system was developed using the RF-DETR model for object detection, Spring Boot for the back-end, and React/Next.js for the front-end, in addition to using PostgreSQL as the database to store application

data. As a result, the automated monitoring made the inspectors' supervision work more efficient, increasing compliance with the law and improving the learning environment.

KEYWORDS: computer vision; automated monitoring; artificial intelligence; object detection; cell phone.

INTRODUÇÃO

O uso de *smartphones* vem sendo adotado amplamente no cotidiano da sociedade moderna, causando impactos contínuos no modo de relacionamento entre os indivíduos. Apesar dos benefícios proporcionados por esses aparelhos às atividades humanas, seu uso indiscriminado pode causar grandes problemas ao desenvolvimento delas, principalmente no que concerne ao uso desses aparelhos nas escolas, causando sintomas como a falta de concentração, a perda das interações sociais e o baixo rendimento acadêmico dos estudantes (Viana *et al.*, 2025). Como resposta a isso, o Governo Federal sancionou a Lei Federal 15.100/2025, que restringe o uso de dispositivos eletrônicos por estudantes do ensino básico dentro das redes de ensino, permitindo-os apenas para atividades pedagógicas (Brasil, 2025). Contudo, a aplicação concreta desta lei encontra sérios empecilhos, uma vez que as instituições de ensino não apresentam recursos de monitoramento, como inspetores de aluno, compatíveis à extensão do espaço escolar para averiguar a enorme quantidade de estudantes ali presentes (Bimbati, 2023).

Nesse sentido, esse projeto visa a criação de um sistema baseado em Visão Computacional (VC), um nicho das Inteligências Artificiais (IAs), capaz de identificar e reportar os estudantes que estiverem utilizando celulares dentro das escolas através das câmeras pré-instaladas nas áreas institucionais em uma interface web para fácil visualização e análise das ocorrências geradas pela IA. Essa interface é alimentada por uma Interface de Programação de Aplicações (*Application Programming Interface* – API), responsável por processar os dados das ocorrências, armazená-los em um banco de dados e, posteriormente, inseri-las no site. Dessa forma, os monitores podem penalizar estudantes descumprindo a lei do uso de celulares de modo facilitado, otimizando sua efetivação.

MATERIAL E MÉTODOS

Inicialmente, foi necessário se aprofundar na temática de desenvolvimento de algoritmos de visão computacional por meio de trabalhos correlatos. Para tanto, realizou-se a leitura de artigos cuja abordagem se assemelhasse ao máximo dos objetivos em questão – a produção de uma IA especializada em identificar pessoas usando celulares. Dentre eles, há de se citar, por conta de sua relevância para o presente trabalho, o artigo de Hatay et al. (2021), que descreve uma solução de detecção de pedestres distraídos em celulares durante a travessia de ruas e rodovias, e o de Moyo et al. (2023), que apresentou uma abordagem para a identificação de alunos usando celulares durante a realização de exames escolares.

Após a aquisição desses conhecimentos iniciais, a atenção foi voltada para a divisão do trabalho em três diferentes frentes, a saber: a construção do conjunto de dados, ou *dataset*, para a aprendizagem da IA seguida de seu treinamento em si; o desenvolvimento da aplicação web para a visualização das ocorrências geradas por ela por parte dos inspetores de alunos; e a codificação da API responsável pela interconexão entre IA e interface web. Na FIGURA 1, é ilustrada uma abstração da estrutura e do fluxo de dados dessa aplicação.

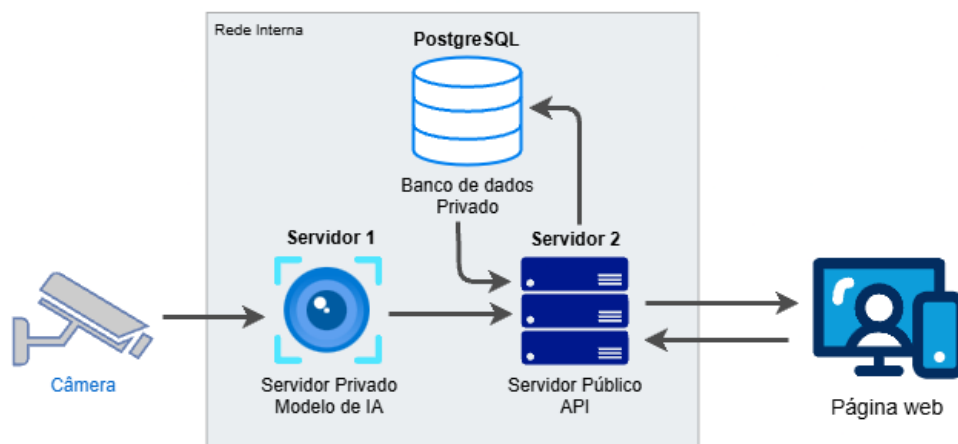


FIGURA 1. Fluxograma da estrutura de funcionamento do sistema e do fluxo de dados da aplicação.

Para a formulação do *dataset*, foram obtidas imagens tanto autorais, capturadas a partir de dispositivos móveis de dentro da escola em que o trabalho foi aplicado (200 imagens), quanto extraídas de repositórios públicos (600 imagens), como o de (TW, 2025) e o de (Rajput, Nag e Mittal, 2020), totalizando 800 imagens. Vale citar que as imagens autorais foram baseadas, exclusivamente, em fotos dos próprios autores do trabalho ou de pessoas que deram consentimento prévio, sem comprometer a ética do direito de imagem de terceiros. Essas imagens foram tratadas na plataforma Roboflow (Roboflow, 2025), que fornece um ambiente completo de desenvolvimento para soluções de visão computacional, através da qual as imagens foram anotadas por meio da classificação dos segmentos contidos nelas de acordo com classes para cada tipo de objeto de interesse - como pessoas usando ou não usando celulares -, conforme ilustrado pela FIGURA 2.

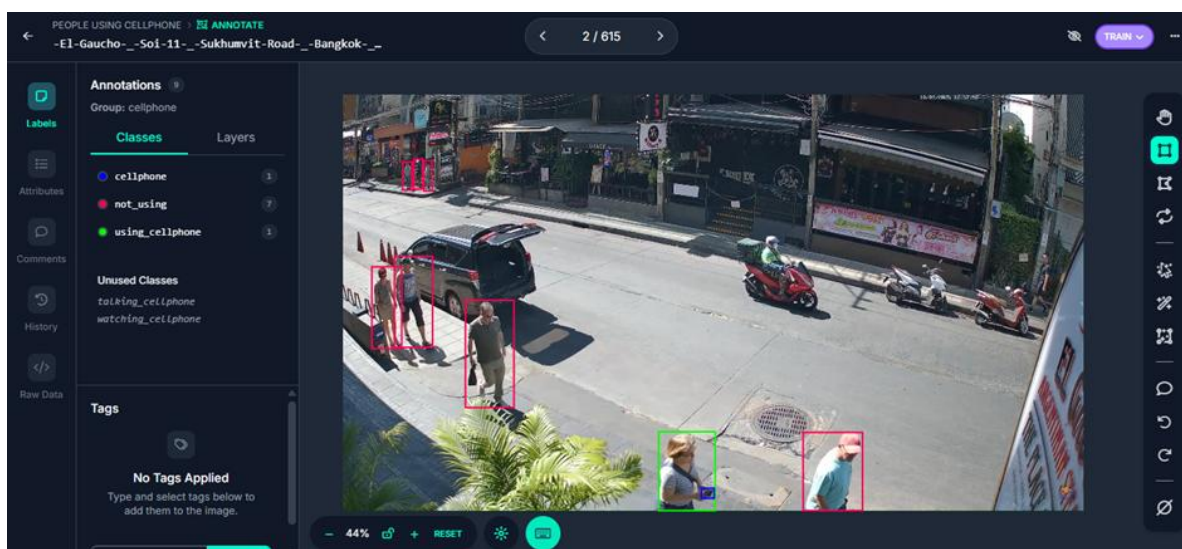


FIGURA 2. Interface de anotação das imagens dentro da plataforma Roboflow com a marcação de uma pessoa usando celular (caixa verde) e de pessoas sem celular (caixas vermelhas).

Ao final da etapa citada acima, foi realizado o treinamento de três modelos de IA distintos com base no mesmo *dataset*, com o fito de analisar a acurácia e eficiência de cada um para posterior implementação na solução final. Esses modelos foram treinados a partir dos algoritmos de Rede Neural Convolucional (*Convolutional Neural Network* – CNN) amplamente conhecidos, como o YOLOv11 (Khanam e Hussain, 2024), Roboflow 3.0 e o RF-DETR. O modelo baseado em YOLOv11n foi treinado localmente através de um computador equipado com um processador Intel I5-11400, 16 GB de memória RAM e uma placa de vídeo Nvidia Geforce RTX 3060, rodando, para o treinamento, um algoritmo feito

na linguagem de programação Python v3.12 (Python, 2025), responsável por realizar o treinamento do modelo com base no *dataset* estipulado. Os demais modelos, ainda, foram treinados utilizando a plataforma Roboflow, sua proprietária e desenvolvedora, já que seu ambiente de desenvolvimento é otimizado para tal fim.

Em seguida, foi dado início ao desenvolvimento da API e da interface web. A API foi desenvolvida por meio da linguagem de programação Java (Java, 2025) em comunhão com o *framework* Spring Boot, responsável por criar *endpoints* pelos quais o site possa acessar e obter os dados das ocorrências, os quais foram armazenados em um banco de dados de tipo relacional conhecido como PostgreSQL (PostgreSQL, 2025). A segurança das rotas da API, por sua vez, foi implementada a partir da biblioteca Spring Security, dos mesmos criadores do Spring Boot (Spring, 2025), garantindo a segurança e integridade dos dados presentes nos processos do sistema.

Finalmente, a aplicação web foi desenvolvida em linguagem de programação TypeScript (TypeScript, 2025), reconhecida pela sua eficiência no tocante à codificação de aplicações web escaláveis e modernas. Para que o TS pudesse ser executado a partir do servidor foi necessário utilizar, junto da biblioteca React e o *framework* Next.js, o Node.js, que adapta tecnologias Javascript para o uso *server-side*.

RESULTADOS E DISCUSSÃO

O sistema desenvolvido apresenta uma interface web com funcionalidades desenvolvidas para ajudar os inspetores na gestão das ocorrências, arranjadas de forma clara e transparente para sua melhor utilização. Nesse contexto, dentre as funcionalidades disponíveis, a mais relevante é a de realizar o *feedback* de uma ocorrência pendente, estabelecida na página principal, como ilustrado na FIGURA 3.

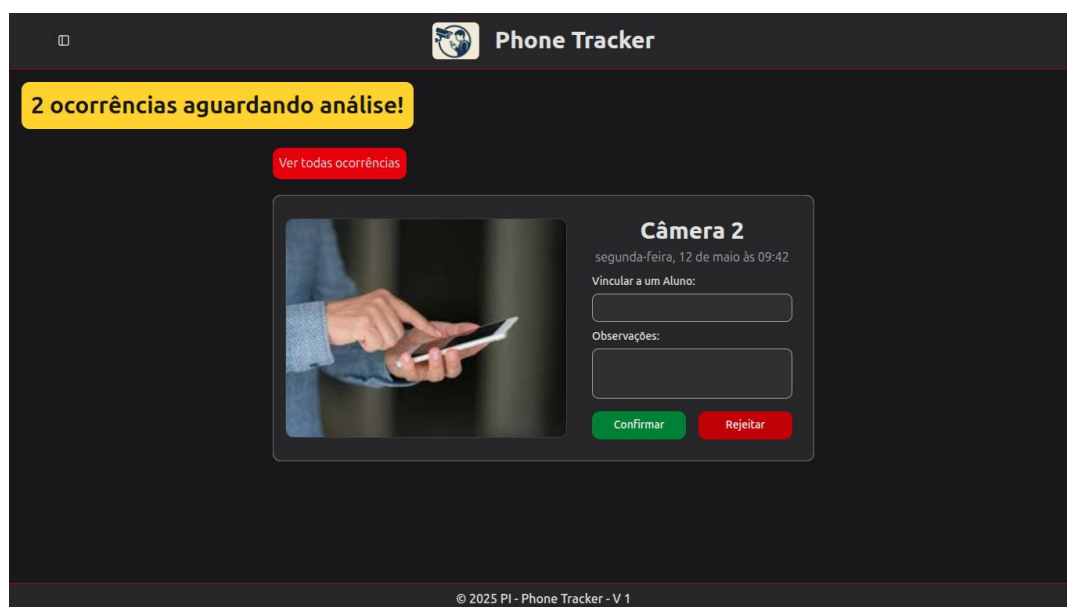


FIGURA 3. Página inicial com ocorrências aguardando avaliação.

Ademais, os modelos de IA gerados a partir das três CNNs anteriormente citadas – YOLO, Roboflow 3.0 e RF-DETR – foram avaliados estatisticamente quanto ao seu desempenho por meio de gráficos para que fosse implementado no sistema aquele com maior eficiência possível. Tais métricas foram a precisão, a precisão média – essa considerando as faixas entre 50% e 95% de intersecção da detecção do modelo –, o *recall* e o *F1-score*, amplamente conhecidas para estudos de desempenho no ramo de *machine learning*. O gráfico gerado com elas é exibido na FIGURA 4, a seguir.

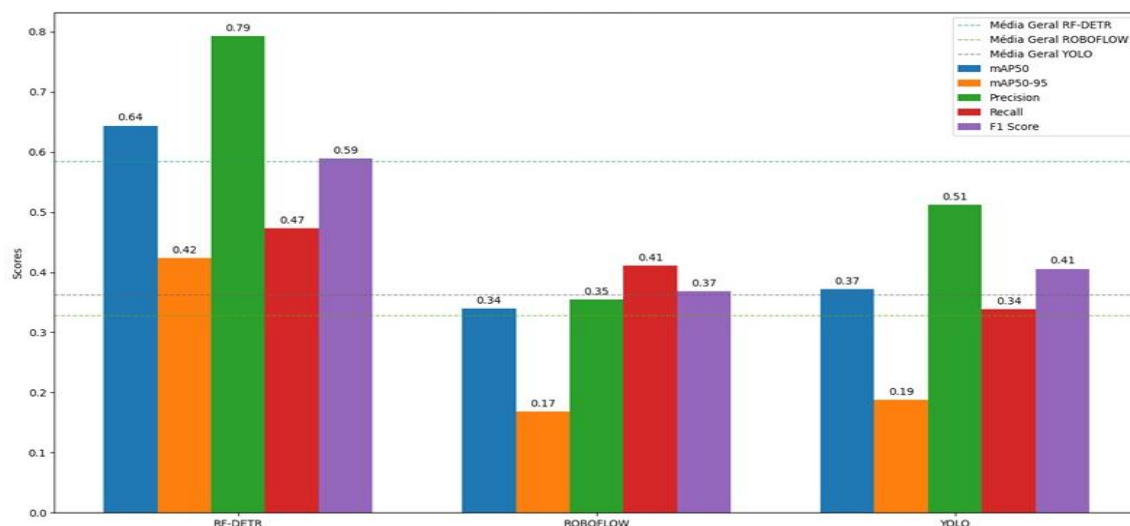


FIGURA 4. Gráfico de desempenho dos modelos por métrica.

Com base nesse gráfico, pode-se notar que a média do desempenho do modelo baseado na CNN RF-DETR é substancialmente superior aos outros ao se comparar métrica a métrica, com uma média geral de 68% de desempenho. Isso se verifica, provavelmente, devido à arquitetura desse algoritmo, pois o torna capaz de analisar integralmente a imagem processada – aprende a classificá-la segundo o contexto –, enquanto os outros realizam esse mesmo processo somente dentro da região que a IA classifica como objeto de interesse. Por isso, o modelo implementado na versão final da aplicação foi o treinado com o RF-DETR.

CONCLUSÕES

Por meio desse sistema foi possível concluir a viabilidade de aplicações integradas com modelos de Visão Computacional para o monitoramento autônomo de alunos usando telefones celulares, possibilitando otimizar o trabalho dos inspetores, tornando os seus trabalhos mais eficazes em combater esses tipos de ações inadequadas, deixando um ambiente de aprendizado mais produtivo. Em trabalhos futuros, almeja-se que seja realizado a expansão e diversificação do *dataset* de treinamento, aliado ao ajuste fino do modelo de IA visando aumentar a precisão e reduzir falsos positivos, aliado de uma reestruturação do algoritmo responsável pela detecção dos infratores, visando aumentar o desempenho da aplicação, principalmente quando utilizada com uma grande quantidade de câmeras.

CONTRIBUIÇÕES DOS AUTORES

T.M.D foi responsável pela prototipação e desenvolvimento da interface web. L.G.S contribuiu com o desenvolvimento da API e com o treinamento dos modelos de IA. G.T.R, J.F.M.B, L.G.S e T.M.D contribuíram para a captura das imagens e para a incorporação destas no *dataset*. G.T.R, J.F.M.B e L.G.S atuaram na redação do trabalho. M.A.T atuou na orientação, revisão crítica e aprovação da versão final.

Todos os autores contribuíram com a revisão do trabalho e aprovaram a versão submetida.

AGRADECIMENTOS

Agradecemos ao IFSP Câmpus Catanduva (CTD) pelo apoio institucional e pela infraestrutura disponibilizada para a realização e execução deste projeto e ao professor orientador dr. Márcio Andrey Teixeira.

REFERÊNCIAS

- BIMBATI, A. P. Sem agentes escolares, diretores acumulam até controle de entrada na escola. **UOL**, 9 abr. 2023. Disponível em: <<https://educacao.uol.com.br/noticias/2023/04/09/falta-agentes-escolares-escola-sp.htm>>. Acesso em: 21 abr. 2025.
- BRASIL. Lei n. 15.100, de 13 de jan. de 2025. **Restrição do Uso de Celulares nas Escolas**, Brasília, DF, 13 jan. 2025.
- HATAY, E. et al. **Learning to Detect Phone-related Pedestrian Distracted Behaviors with Synthetic**. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. [S.l.]: [s.n.]. 2021. p. 2981-2989.
- JAVA. **Java 24**, 2025. Disponível em: <<https://dev.java/>>. Acesso em: 30 jul. 2025.
- KHANAM, R.; HUSSAIN, M. **Yolov11: An overview of the key architectural enhancements**. arXiv. [S.l.]. 2024. (2410.17725).
- MOYO, R. et al. **Using CNN to Detect the Use of Mobile Phone in Examination Rooms**. DeepLearningIndaba Workshop. Online: OpenReview. 2023.
- POSTGRESQL. PostgreSQL: The World's Most Advanced Open Source Relational Database, 2025. Disponível em: <<https://www.postgresql.org/>>. Acesso em: 30 jul. 2025.
- PYTHON. **Linguagem de programação Python**, 2025. Disponível em: <<https://www.python.org/>>. Acesso em: 02 jun. 2025.
- RAJPUT, P.; NAG, S.; MITTAL, S. Detecting usage of mobile phones using deep learning technique. **Proceedings of the 6th EAI International Conference on Smart Objects and Technologies for Social Good**, Antwerp, Belgium, jul. 2020. 96–101.
- ROBOFLOW. Roboflow: Computer vision tool. **Roboflow**, 2025. Disponível em: <<https://roboflow.com/>>. Acesso em: 07 jul. 2025.
- SPRING. **Spring Boot 3.5.0**, 2025. Disponível em: <<https://spring.io/projects/spring-boot>>. Acesso em: 05 jun. 2025.
- TW. Phone 2 Dataset. **Roboflow Universe**, 2025. Disponível em: <<https://universe.roboflow.com/tw-qoffq/phone-2-70mlo>>. Acesso em: 02 ago. 2025.
- TYPESCRIPT. **Documentação do TypeScript**, 2025. Disponível em: <<https://www.typescriptlang.org/docs/>>. Acesso em: 30 jul. 2025.
- VIANA, H. et al. Distração ou recurso pedagógico? Como especialistas avaliam o uso de celulares nas escolas; veja perguntas e respostas. **G1**, 29 jan 2025. Disponível em: <<https://g1.globo.com/educacao/noticia/2025/01/29/distracao-ou-recurso-pedagogico-como-especialistas-avaliam-o-uso-de-celulares-nas-escolas-veja-perguntas-e-respostas.ghml>>. Acesso em: 04 abr. 2025.