

Reconhecimento Visual de Algoritmos Físicos como Ferramenta de Ensino de Programação para Crianças

Ana Mara de Oliveira Figueiredo, Guilherme Amaral Ribas da Costa,
Felipe Basílio Alves, Ianne Lima Nogueira

¹Instituto Federal Fluminense Campos Bom Jesus do Itapaboana (IFF)
Bom Jesus do Itapaboana – RJ – Brasil

{ana.figueiredo, guilherme.ribas}@gsuite.iff.edu.br

{f.basilio, ianne.nogueira}@gsuite.iff.edu.br

Abstract. Introduction: Expanding access to programming education has proven essential for reducing inequalities and promoting digital inclusion from early childhood, especially in socially vulnerable contexts. **Objective:** This article presents an educational tool for teaching programming to children in socially vulnerable situations, combining physical MDF pieces and computer vision. **Steps:** Children build concrete algorithms using physical pieces, which are captured by image and automatically converted into Python code using OpenCV. **Expected Results:** The proposal encourages computational literacy, logical thinking, and technological engagement in a playful and collaborative way, promoting digital inclusion and reducing educational inequalities. **Keywords** Digital Inclusion, Programming Education, Computational Literacy, Computer Vision.

Resumo. Introdução: A ampliação do acesso à educação em programação tem se mostrado essencial para reduzir desigualdades e promover a inclusão digital desde a infância, especialmente em contextos de vulnerabilidade social. **Objetivo:** Este artigo apresenta uma ferramenta educacional para ensino de programação a crianças em vulnerabilidade social, combinando peças físicas de MDF e visão computacional. **Etapas:** As crianças montam algoritmos concretos que são capturados por imagem e convertidos em código Python automaticamente, usando OpenCV. **Resultados esperados:** A proposta incentiva o letramento computacional, o pensamento lógico e o engajamento tecnológico de forma lúdica e colaborativa, promovendo inclusão digital e redução das desigualdades educacionais. **Palavras-Chave** Inclusão Digital, Ensino de Programação, Letramento Computacional, Visão Computacional.

1. Introdução

O acesso equitativo à educação tecnológica permanece como um dos principais desafios educacionais do século XXI. Em uma sociedade cada vez mais orientada por dados, automação e inteligência artificial, o desenvolvimento de competências digitais tornou-se indispensável para a formação cidadã e a participação plena no mundo contemporâneo. No entanto, estudantes de comunidades em situação de vulnerabilidade

social frequentemente enfrentam barreiras no acesso a recursos didáticos, tecnologias assistivas e metodologias de ensino adequadas.

Neste contexto, este artigo apresenta uma solução inovadora e acessível: uma ferramenta pedagógica baseada em visão computacional que permite a leitura de algoritmos físicos construídos por crianças com peças de MDF ou papel grosso, convertendo-os automaticamente em código Python. As peças, feitas de forma padronizada, representam comandos de uma linguagem de programação em blocos. A manipulação concreta desses blocos permite que os alunos construam algoritmos de maneira tangível, visual e colaborativa, favorecendo a compreensão de estruturas como sequência, repetição e condição lógica.

Ao integrar tecnologias emergentes — como o reconhecimento de padrões por imagem e a geração automatizada de código — com uma proposta lúdica e inclusiva, o projeto busca promover o letramento computacional desde os anos iniciais da escolarização. A abordagem contribui tanto para a alfabetização tecnológica, quanto para o desenvolvimento do pensamento lógico, da criatividade e da autonomia dos estudantes.

Além de seu valor educacional, a proposta possui relevância social e científica. No campo da ciência da computação, avança ao explorar aplicações de visão computacional em contextos educativos. No campo social, oferece uma alternativa concreta para a inclusão digital, especialmente em regiões com infraestrutura limitada.

2. Fundamentação teórica

O pensamento computacional é amplamente reconhecido como uma habilidade essencial a ser desenvolvida desde os primeiros anos da educação básica, pois envolve competências como resolução de problemas, raciocínio lógico e criatividade [Wing 2006]. Abordagens pedagógicas que incorporam atividades lúdicas e materiais manipulativos estão alinhadas às teorias construtivistas, que defendem o aprendizado por meio da experimentação e da construção ativa do conhecimento [Papert 1980] e [Resnick 2017]. Nesse sentido, a cultura maker e a aprendizagem baseada em projetos ganham destaque por promoverem o protagonismo infantil, a autonomia e o engajamento significativo no processo de aprendizagem [Martinez e Stager 2013], aspectos essenciais quando se busca ensinar programação a crianças, especialmente em contextos de vulnerabilidade social.

Dentro desse cenário, o ensino de algoritmos ocupa um papel central. Algoritmos são sequências ordenadas de instruções voltadas à solução de problemas ou à realização de tarefas, sendo a base do pensamento computacional. Compreendê-los permite que as crianças desenvolvam habilidades de decomposição, abstração e lógica, fundamentais não apenas para programar, mas também para resolver problemas em outras áreas do conhecimento [De Oliveira et al. 2014]. Quando explorados por meio de recursos concretos — como peças físicas coloridas —, os algoritmos tornam-se mais acessíveis e compreensíveis para o público infantil, contribuindo para um aprendizado mais ativo, visual e inclusivo. Essa é justamente a proposta do presente trabalho, que utiliza algoritmos físicos como meio de aprendizagem tangível e interativa.

Uma abordagem pedagógica que tem se mostrado eficaz nesse contexto é a programação em blocos. Essa metodologia utiliza elementos visuais — blocos que representam comandos e estruturas — que podem ser encaixados para formar

algoritmos completos [de Lima Sousa et al. 2020]. Plataformas como Scratch e Blockly exemplificam essa abordagem, facilitando o entendimento dos conceitos computacionais ao eliminar barreiras relacionadas à sintaxe textual. Para crianças em fase de alfabetização ou com menor familiaridade com tecnologia, a programação em blocos promove uma experiência mais intuitiva, divertida e exploratória. A proposta deste artigo se inspira nesse modelo visual, porém introduz um novo nível de interatividade ao permitir que blocos físicos montados pelas crianças sejam interpretados por um sistema computacional.

Na vertente tecnológica, a visão computacional tem possibilitado a criação de interfaces naturais entre o mundo físico e os sistemas digitais, permitindo que dispositivos reconheçam, interpretem e respondam a imagens do ambiente real [Szeliski 2010]. Com bibliotecas como o OpenCV [Bradski 2000], é possível desenvolver sistemas capazes de realizar tarefas complexas de reconhecimento de padrões e objetos em tempo real — uma característica essencial para aplicações educacionais interativas como a proposta deste trabalho. A combinação entre elementos físicos e interpretação visual automatizada representa uma oportunidade promissora para tornar o aprendizado mais inclusivo e dinâmico.

Mais especificamente, o reconhecimento de imagens, subárea da visão computacional, é responsável por identificar e interpretar padrões visuais em fotografias ou vídeos. De acordo com [Gonzalez e Woods 2000], esse processo envolve etapas como aquisição da imagem, pré-processamento (ajustes de brilho, contraste, remoção de ruído), segmentação de elementos de interesse, extração de características e classificação. No caso do presente projeto, essas etapas são aplicadas para identificar a disposição e a cor das peças físicas montadas pelas crianças, reconhecendo-as como estruturas algorítmicas.

Por fim, o uso do Python como linguagem de programação para conversão dos algoritmos físicos é uma escolha estratégica. Segundo [Borges 2014], Python destaca-se por sua sintaxe simples, clareza semântica e ampla adoção na comunidade educacional e científica. Além disso, oferece suporte a diversas bibliotecas voltadas para ciência de dados, inteligência artificial e visão computacional — como NumPy, Pandas, TensorFlow e OpenCV —, facilitando o desenvolvimento de soluções robustas e integradas. No contexto do projeto, a geração automática de código Python a partir de algoritmos físicos proporciona uma ponte concreta entre o mundo real e o digital, permitindo que as crianças vejam, em tempo real, o resultado computacional das sequências que construíram fisicamente, o que reforça a aprendizagem significativa e fortalece a alfabetização computacional de forma acessível e lúdica.

3. Metodologia

O presente projeto foi desenvolvido com o objetivo de criar uma ferramenta educacional baseada em visão computacional capaz de interpretar algoritmos físicos, construídos com peças de MDF ou papel grosso, e convertê-los automaticamente em código Python.

A primeira etapa consistiu no design e produção das peças físicas, em que cada peça representa um comando básico de programação — como início, fim, movimento, condição e repetição — e foi padronizada com simbologias visuais e cores distintas. Essa padronização facilita tanto a compreensão por parte das crianças quanto a detecção automatizada pelo sistema. A escolha do MDF e do papel grosso justifica-

se pela combinação de baixo custo, resistência e facilidade de manuseio, características adequadas para o contexto educacional.

Na segunda etapa, estruturou-se o ambiente educativo para a aplicação das oficinas pedagógicas em escolas públicas e projetos sociais parceiros. Durante essa fase, foram realizados testes iniciais com diferentes formas de montagem dos algoritmos físicos, acompanhados da captura fotográfica dos arranjos, que servirão para o processamento posterior das imagens. Nessas oficinas, as crianças serão incentivadas a montar algoritmos físicos utilizando as peças sobre superfícies planas, com o suporte contínuo de bolsistas do projeto. Todas as interações serão registradas fotograficamente, respeitando os critérios éticos, sobretudo no que tange à proteção da imagem de menores de idade.

A terceira etapa, ainda não implementada, prevê a coleta e rotulagem das imagens obtidas durante as oficinas, visando a construção de um banco de dados diversificado, contendo diferentes composições das peças em variadas condições de iluminação, posicionamento e fundo. Com esse banco, será desenvolvido o sistema de visão computacional utilizando a biblioteca OpenCV. Serão aplicadas técnicas de pré-processamento de imagens, segmentação por cor, detecção de contornos e reconhecimento de padrões, com o objetivo de identificar a sequência das peças montadas e interpretá-las como algoritmos visuais. Em seguida a implementação do módulo de tradução, que converterá a sequência reconhecida das peças em código Python. O código gerado será exibido em tempo real por meio de uma interface amigável.

4. Resultados iniciais

A primeira fase do projeto resultou na produção de 50 peças de MDF/Papel grosso representando comandos básicos da lógica de programação e na preparação de oficinas educativas para crianças, com idades entre 8 e 12 anos, em escolas públicas parceiras. Para essas oficinas, foram montados 15 algoritmos físicos utilizando as peças coloridas, cujas composições foram registradas fotograficamente. As imagens capturadas estão sendo organizadas e rotuladas manualmente para compor um banco de dados que servirá de base para o desenvolvimento do sistema de visão computacional.

Mesmo em fase inicial, foi possível observar o potencial pedagógico da proposta. Os bolsistas envolvidos também relataram ganhos significativos em habilidades técnicas e interpessoais, como comunicação, organização e adaptação ao ambiente escolar. O interesse das escolas parceiras reforça a relevância da iniciativa e seu potencial de continuidade e ampliação. A Figura 1 ilustra um exemplo de algoritmo físico construído.

5. Conclusão

A proposta apresentou resultados promissores ao integrar blocos físicos e conceitos básicos de programação, oferecendo uma alternativa acessível e concreta para a introdução ao pensamento computacional no ensino fundamental.

Nos próximos passos, serão implementadas as etapas de processamento visual e tradução automática para código Python. Também está prevista a ampliação do banco de dados de imagens e o refinamento da interface do sistema. Com isso, espera-se consolidar uma ferramenta de baixo custo, replicável e com forte potencial de impacto educacional, especialmente em contextos com menos acesso à tecnologia.

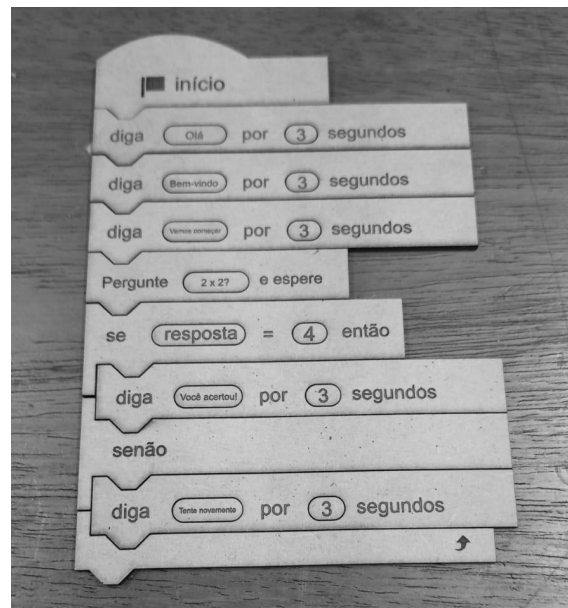


Figura 1. Algoritmo físico montado representando comandos básicos de programação.

6. Questões éticas

Este estudo está sendo conduzido em conformidade com as diretrizes estabelecidas pelo Comitê de Ética em Pesquisa com Seres Humanos do Instituto Federal Fluminense (IFFluminense), conforme informações disponíveis no portal institucional (<https://portal1.iff.edu.br/o-iffuminense/pesquisa/comite-de-etica-em-pesquisa-com-seres-humanos-cep-iffuminense>) e em atendimento às Normas e Diretrizes Regulamentadoras da Pesquisa Envolvendo Seres Humanos (Resoluções CNS n.º 466/2012, 510/2016 e 674/2022, entre outras aplicáveis).

Todas as atividades estão sendo realizadas no ambiente institucional, com a participação voluntária de estudantes e docentes. Para participantes menores de idade, foi obtida autorização prévia dos responsáveis legais por meio do Termo de Consentimento Livre e Esclarecido (TCLE). Não foram coletados dados pessoais que permitissem a identificação direta ou indireta dos participantes, assegurando a privacidade e o anonimato.

Os dados obtidos são utilizados exclusivamente para fins acadêmicos e científicos, armazenados de forma segura e não compartilhados com terceiros fora do escopo do projeto. O estudo foi planejado e está sendo conduzido de modo a minimizar riscos, prevenir impactos negativos aos participantes e assegurar transparência e reprodutibilidade dos resultados.

7. Declaração sobre o uso de ferramentas de IA

As pessoas autoras declaram que não utilizaram ferramentas de inteligência artificial generativa em nenhuma etapa da concepção, desenvolvimento, redação ou revisão deste artigo. Todo o conteúdo técnico e científico foi integralmente produzido pelas autoras e revisado manualmente. Todas as ideias, análises, resultados e conclusões aqui apresentados são de responsabilidade exclusiva das pessoas autoras.

Referências

- Borges, L. E. (2014). *Python para desenvolvedores: aborda Python 3.3*. Novatec Editora.
- Bradski, G. (2000). The opencv library. *Dr. Dobb's Journal of Software Tools*.
- de Lima Sousa, L., Farias, E. J., e de Carvalho, W. V. (2020). Programação em blocos aplicada no ensino do pensamento computacional: Um mapeamento sistemático. In *Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 1513–1522. SBC.
- De Oliveira, M., De Souza, A., Ferreira, A., e Barreiros, E. (2014). Ensino de lógica de programação no ensino fundamental utilizando o scratch: um relato de experiência. In *Workshop sobre Educação em Computação (WEI)*, pages 239–248. SBC.
- Gonzalez, R. C. e Woods, R. E. (2000). *Processamento de imagens digitais*. Editora Blucher.
- Martinez, S. L. e Stager, G. (2013). *Invent To Learn: Making, Tinkering, and Engineering in the Classroom*. Constructing Modern Knowledge Press.
- Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.
- Resnick, M. (2017). *Lifelong Kindergarten: Cultivating Creativity Through Projects, Passion, Peers, and Play*. MIT Press.
- Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.