

UNIVERSIDADE FEDERAL DE ITAJUBÁ

Detecção de Câncer Cervical

André Santos Cruz

Felipe Emanuel Pereira Gonçalves

Vivian Leite Fragoso

ITAJUBÁ

2025

1. Introdução.....	3
2. Motivação.....	3
3. Objetivos.....	4
4. Hardware utilizado.....	4
5. Metodologia.....	6
5.1 Coleta de dados.....	6
5.2 Definição de parâmetros.....	8
5.3 Treinamento do Modelo.....	10
6. Resultados.....	12
7. Testes.....	13
7.1 Teste de Modelo.....	13
7.2 Teste Prático.....	15
8. Conclusão.....	17
9. Referências.....	18

1. Introdução

A detecção precoce do câncer cervical ainda é um grande desafio em países em desenvolvimento, onde a escassez de profissionais especializados limita o acesso a exames de qualidade. Nesse contexto, soluções baseadas em inteligência artificial (IA) emergem como uma alternativa promissora, capaz de ampliar o alcance da triagem, reduzir custos e padronizar os diagnósticos, tornando a prevenção mais eficiente e acessível.

O uso de tecnologias embarcadas, como o EdgeML, permite que modelos de aprendizado de máquina sejam executados diretamente em dispositivos de baixo custo e baixo consumo de energia. Isso viabiliza a análise automatizada de exames em regiões remotas ou com infraestrutura limitada, garantindo que mais mulheres tenham acesso a avaliações precisas e rápidas.

Além dos benefícios clínicos, a automação da triagem contribui para sistemas de saúde mais organizados e resilientes. Processos padronizados diminuem erros humanos, agilizam o fluxo de atendimento e possibilitam a integração de dados em larga escala, fortalecendo políticas de prevenção e monitoramento populacional.

Este relatório apresenta o desenvolvimento de um sistema embarcado para detecção de alterações cervicais precoces utilizando EdgeML, detalhando as etapas de preparo do dataset, treinamento do modelo e implementação do sistema em dispositivo embarcado. Os testes foram realizados em cenário controlado, com imagens previamente anotadas e processadas, sem envolvimento direto de unidades de saúde ou coleta de novos dados. A proposta busca demonstrar o potencial da inteligência artificial embarcada para apoiar o diagnóstico precoce e ampliar o acesso a tecnologias de rastreamento preventivo.

2. Motivação

A motivação deste projeto é aplicar, de forma prática, os conhecimentos adquiridos na disciplina IESTI05 para desenvolver uma solução tecnológica voltada à detecção precoce do câncer cervical. Em muitos países em desenvolvimento, a falta de profissionais especializados e o acesso limitado a exames contribuem para diagnósticos tardios, reduzindo significativamente a eficácia da prevenção. Ao utilizar EdgeML, é possível criar dispositivos capazes de realizar triagens automatizadas diretamente em unidades de saúde ou programas comunitários, ampliando o acesso a cuidados preventivos mesmo em regiões com recursos restritos.

Os principais objetivos deste projeto são: automatizar a detecção de alterações cervicais a partir da análise de imagens; desenvolver uma solução acessível, portátil e de baixo consumo energético utilizando aprendizado de máquina embarcado; e testar sua aplicabilidade em cenários reais, como clínicas móveis, unidades de saúde comunitárias e programas de rastreamento populacional.

3. Objetivos

O objetivo deste projeto é desenvolver um modelo de EdgeML capaz de identificar automaticamente lesões cervicais a partir de imagens médicas, apoiando o diagnóstico precoce do câncer cervical e ampliando o acesso a exames de qualidade em regiões com recursos limitados. A proposta busca combinar inovação tecnológica e impacto social, tornando o rastreamento preventivo mais acessível, ágil e padronizado.

Entre os objetivos específicos, destacam-se:

- Aplicar os conhecimentos adquiridos na disciplina IESTI05 – Machine Learning Systems Engineering no desenvolvimento de um sistema capaz de detectar alterações cervicais de forma autônoma;

- Investigar e implementar técnicas de machine learning otimizadas para execução em dispositivos de borda, com foco em eficiência energética e portabilidade;
- Apoiar o diagnóstico precoce e reduzir a dependência de especialistas, viabilizando o uso do sistema em unidades de saúde comunitárias e programas de triagem em larga escala.

4. Hardware utilizado

Para a implementação do sistema, foi utilizada a Raspberry Pi Zero 2 W, escolhida pelo baixo consumo e tamanho compacto. A placa possui processador quad-core ARM Cortex-A53 (1 GHz), 512 MB de RAM e conectividade Wi-Fi 2.4 GHz e Bluetooth 4.2, sendo adequada para aplicações embarcadas leves. O sistema operacional Raspberry Pi OS Lite (64 bits) foi configurado em modo headless com acesso via SSH.

A Raspberry Pi Camera Module 2, com sensor Sony IMX219 de 8 MP, foi utilizada apenas para inferência local, permitindo testar o modelo de detecção de lesões cervicais em tempo real no dispositivo.

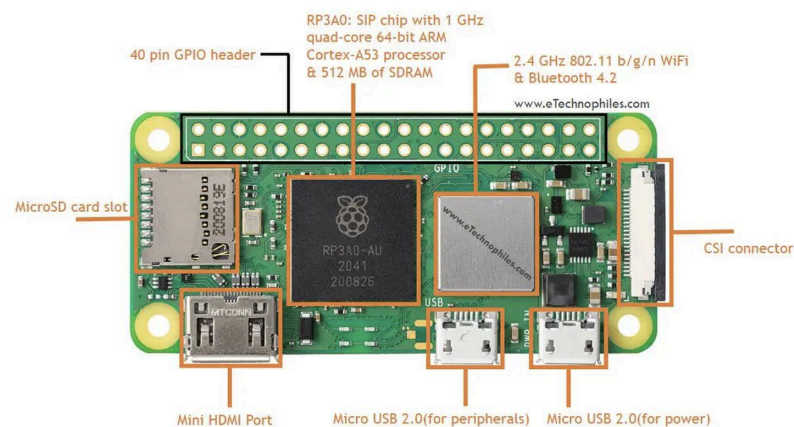


Figura 1 - Placa Raspberry Pi 2W

5. Metodologia

5.1 Dataset

As imagens utilizadas foram selecionadas a partir do *IARC Cervical Cancer Image Bank*, que contém cerca de 200 casos e 913 imagens de colposcopia. Com isso foi feita a seleção e anotação manual realizadas na plataforma *Roboflow*, distribuídas em três classes:

Classe	Descrição	Dados Coletados
LSIL	Lesões intraepiteliais escamosas de baixo grau	28 itens
HSIL	Lesões intraepiteliais escamosas de alto grau	33 itens
Normal	Imagens sem alterações visíveis	29 itens
Total	Após data augmentation	211 itens

Tabela 1 - Descrição das imagens coletadas.

Essa limitação foi adotada porque havia poucas amostras da classe LSIL, e o objetivo era evitar desbalanceamento entre as categorias. O processo de data augmentation aplicou transformações como espelhamento horizontal e vertical, além de rotações de 90°, 180° e 270°, gerando novas variações das imagens. Após o aumento dos dados, o conjunto final passou a ter 211 imagens, divididas em 87% para treinamento (184), 9% para validação (18) e 4% para teste (9), todas redimensionadas para 640×640 px.

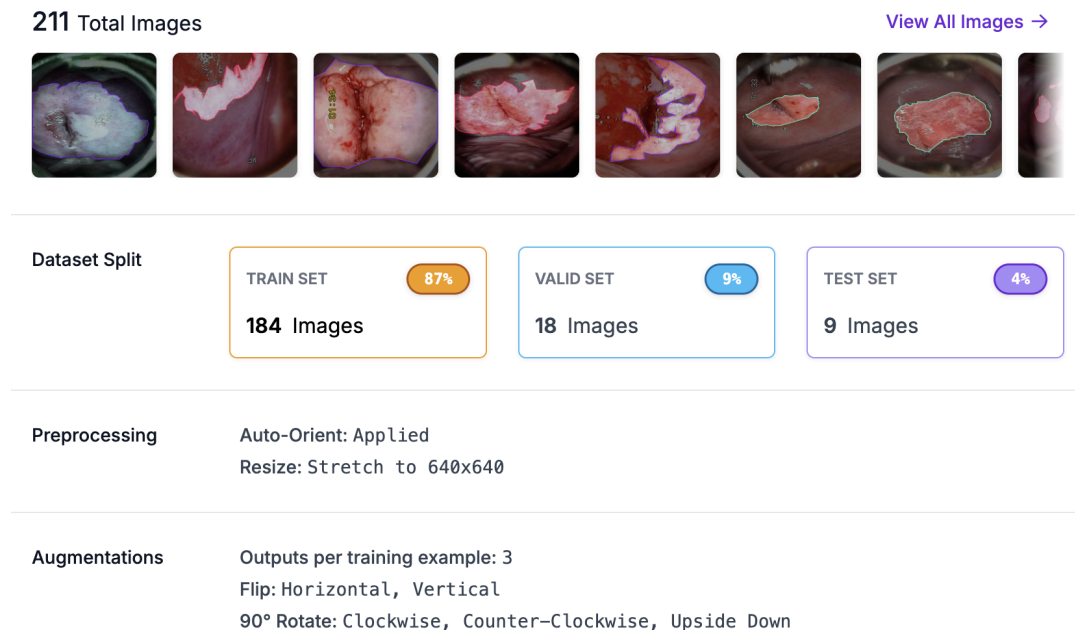


Figura 2 - Versão final da base de dados

5.2 Treinamento do Modelo

Para o treinamento foi utilizada a arquitetura YOLO11n, uma rede neural otimizada para alta eficiência e baixo custo computacional. Essa versão foi escolhida por apresentar um bom equilíbrio entre velocidade de inferência, uso de memória e precisão, sendo adequada para execução em dispositivos embarcados como a Raspberry Pi Zero 2 W.

O modelo foi configurado com as seguintes especificações:

- Tamanho de entrada: 640×640 pixels;
- Número de épocas: 100;
- Taxa de aprendizagem: padrão da biblioteca Ultralytics YOLO;
- Processamento: realizado no ambiente Google Colab com GPU Tesla T4.

6. Resultados

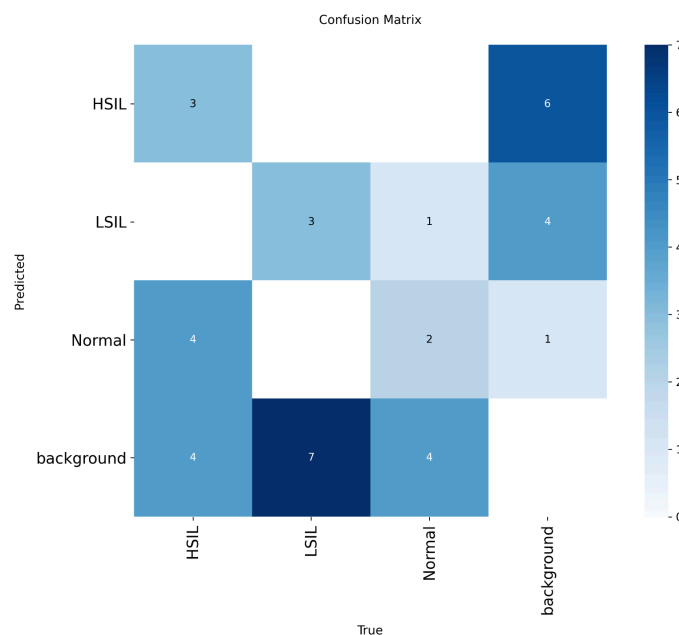


Figura 3 - Resultados do treinamento

Os resultados indicaram precisão média (P) de 0,42, recall (R) de 0,31 e média da precisão (mAP@50) de 0,304, com desempenho semelhante entre as classes: HSIL (0,348), LSIL (0,337) e Normal (0,229). O valor global de mAP@50–95 foi 0,201, refletindo o desafio de distinguir lesões com características visuais semelhantes.

A matriz de confusão mostra que o modelo apresentou confusão entre HSIL, LSIL e Normal, além de algumas previsões incorretas como background, indicando baixa sensibilidade para lesões leves e tendência a falsos positivos. Apesar dessas limitações, o tempo médio de inferência foi de 3,4 ms por imagem, comprovando que o modelo é leve e adequado para execução em dispositivos embarcados, como a Raspberry Pi Zero 2 W.

7. Testes

7.1 Teste do Modelo

Após o treinamento, o modelo YOLO11n foi testado com 9 imagens do conjunto de teste, utilizando limiar de confiança de 0,25. O tempo médio de execução foi de 1,6 ms no pré-processamento, 8,6 ms na inferência e 1,6 ms no pós-processamento por imagem.

Apesar dos resultados limitados observados na matriz de confusão e nas métricas finais de treinamento, o modelo apresentou bom desempenho nos testes, acertando 6 das 9 imagens e identificando corretamente as classes HSIL, LSIL e Normal.

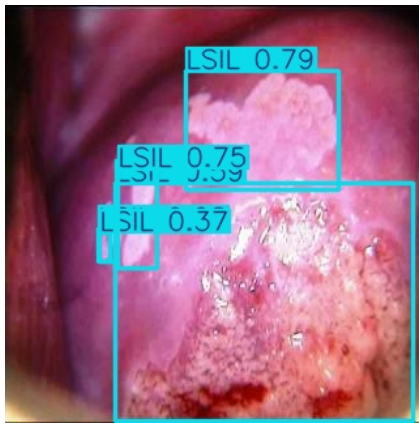


Figura 4 - Resultados de Teste para LSIL

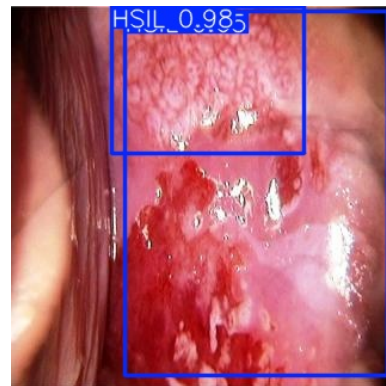


Figura 5 - Resultados de Teste para HSIL

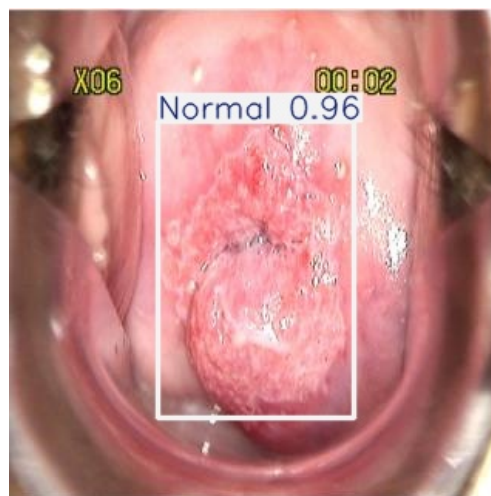


Figura 6 - Resultados de Teste para Normal

7.2 Teste embarcado

O modelo também foi testado diretamente na Raspberry Pi Zero 2 W, realizando inferência com imagens do próprio dataset e tentativas de detecção em tempo real utilizando a câmera embarcada. No entanto, não houve detecção das lesões durante as inferências, tanto nas imagens estáticas quanto nas capturas ao vivo.

```
[21]: model_path = "./models/best.pt"

[27]: capture_image(img_path)
infer_yolo("./images/cam_img_yolo_test.jpg")

[0:24:03.853896538] [1455] INFO Camera camera_manager.cpp:326 libcamera v0.5.1+100-e53bdf1f
[0:24:03.919611756] [1467] WARN RPISdn sdn.cpp:40 Using legacy SDN tuning - please consider moving SDN inside rpi.denoise
[0:24:03.924989921] [1467] INFO RPI vc4.cpp:440 Registered camera /base/soc/l2c0mux/l2c@1/ov5647@36 to Unicam device /dev/media0 and ISP device /dev/media0
[0:24:03.925105702] [1467] INFO RPI pipeline_base.cpp:1107 Using configuration file '/usr/share/libcamera/pipeline/rpi/vc4/rpi_apps.yaml1'
[0:24:03.947313934] [1455] INFO Camera camera.cpp:1205 configuring streams: (0) 320x320-BGR888/sr08 (1) 640x480-SGBRG10_CS12P/RAU
[0:24:03.948311431] [1467] INFO RPI vc4.cpp:615 Sensor: /base/soc/l2c0mux/l2c@1/ov5647@36 - Selected sensor format: 640x480-SGBRG10_1X10 - Selected un
icam format: 640x480-pGAA
Image captured: ./images/cam_img_yolo_test.jpg

Image 1/1 /home/felipe/project/images/cam_img_yolo_test.jpg: 640x640 (no detections), 3801.3ms
Speed: 29.8ms preprocess, 3801.3ms inference, 6.2ms postprocess per image at shape (1, 3, 640, 640)

YOLO Result
```

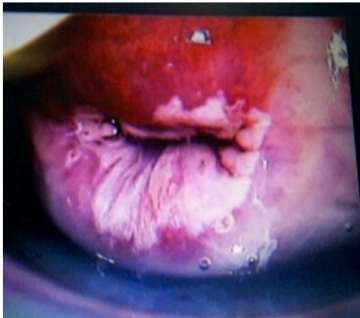


Figura 7 - Resultado da inferência com a câmera

Durante os testes, observou-se um uso intenso dos recursos do sistema, conforme mostrado na Figura 8. O processo principal do Python manteve o uso de CPU acima de 140% (equivalente a quase dois núcleos em uso contínuo) e o consumo de memória próximo de 250 MB, com swap atingindo cerca de 400 MB. Esses valores indicam que o modelo foi executado no limite da capacidade da Raspberry Pi Zero 2 W, ocasionando lentidão e instabilidade durante a inferência.

Apesar das limitações de desempenho, o teste demonstrou a viabilidade da execução local de modelos de visão computacional, reforçando a necessidade de otimizações adicionais ou

o uso de versões quantizadas e mais leves para garantir detecção em tempo real em dispositivos embarcados de baixo custo.

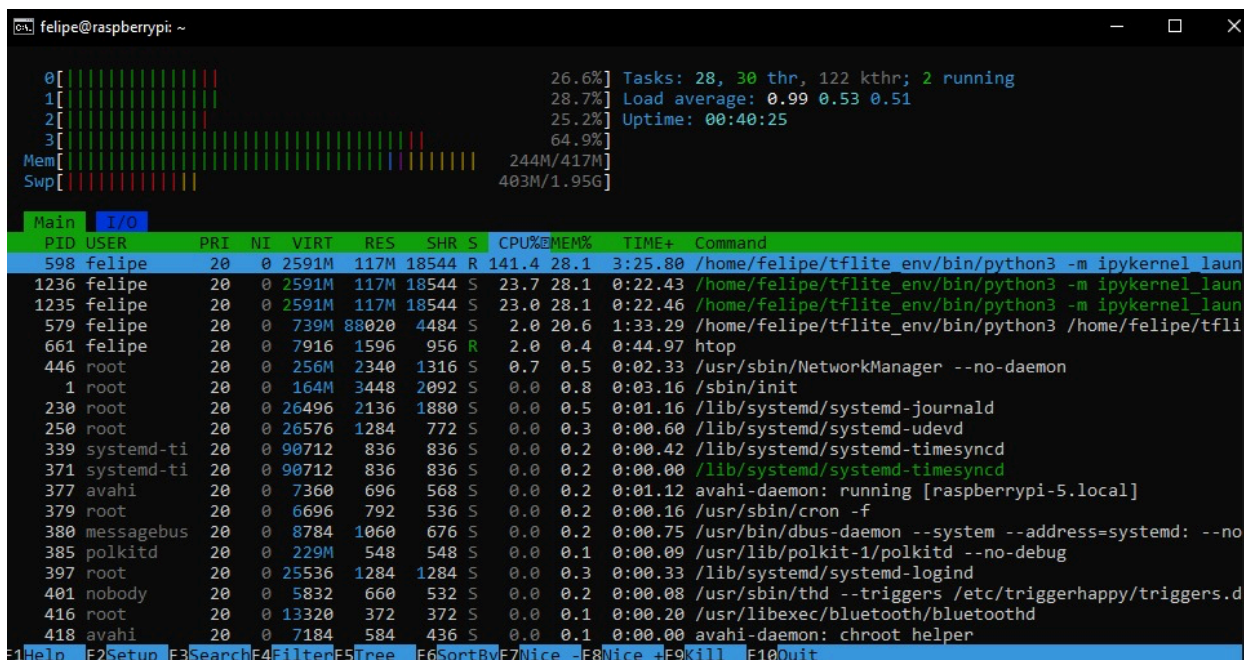


Figura 8 - Uso de recursos

8. Conclusão

O projeto demonstrou a viabilidade de aplicar técnicas de inteligência artificial embarcada (EdgeML) para a detecção automática de lesões cervicais em imagens de colposcopia, utilizando a arquitetura YOLO11n e a Raspberry Pi Zero 2 W como plataforma de teste. O modelo alcançou média de precisão de 0,304 e mostrou capacidade de identificar padrões relevantes, embora com confusões frequentes entre as classes HSIL, LSIL e Normal, evidenciadas na matriz de confusão.

Durante os testes, o modelo apresentou bom comportamento nas inferências em ambiente controlado, acertando 6 das 9 imagens do conjunto de teste, mas não conseguiu realizar detecções consistentes quando executado diretamente na Raspberry Pi, devido às limitações

computacionais. O uso de recursos observou CPU acima de 140% e memória próxima de 250 MB, indicando execução no limite da capacidade do hardware.

Conclui-se que, embora os resultados sejam promissores para validação conceitual, o desempenho do modelo pode ser aprimorado com um dataset mais extenso e variado, contendo anotações mais precisas das diferenças visuais entre os graus de lesão. Essa melhoria é essencial para reduzir as confusões entre classes e aumentar a sensibilidade e precisão do sistema, tornando-o mais robusto para aplicações reais de triagem automatizada e apoio ao diagnóstico precoce do câncer cervical.

9. Referências

INTERNATIONAL AGENCY FOR RESEARCH ON CANCER (IARC). Cervical Cancer Image Bank. Lyon: World Health Organization, 2024. Disponível em: <https://screening.iarc.fr/cervicalimagebank.php>. Acesso em: 22 out. 2025.

DAYALANE, Sundaranarayana et al. Cervical Cancer Classification Using Deep Learning Approach Using Colposcopy Images. *Neural Processing Letters*, v. 57, p. 65, 2025. DOI: 10.1007/s11063-025-11770-w.

SHA, Yuyang et al. CerviFusionNet: A Multi-modal, Hybrid CNN-Transformer-GRU Model for Enhanced Cervical Lesion Multi-classification. *iScience*, v. 27, n. 111313, p. 1–8, dez. 2024. DOI: 10.1016/j.isci.2024.111313.