

Machine Learning para Detecção de Fogo e Fumaça: Uma Metodologia de Trabalho com YOLOv11

Rafael Melchior de Oliveira¹, Lucas Ribeiro Luchesi¹,
Arthur Octávio Oliveira Confessor¹, Tiago de Almeida Lacerda¹,
Pedro Clarindo da Silva Neto¹, Evandro César Freiburger¹

¹Departamento de Computação
Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso (IFMT)
Campus Cuiabá - Cel. Octayde Jorge da Silva
Rua Zulmira Canavarros, 95 - Centro, Cuiabá - MT, Brazil
{r.melchior, arthur.confessor}@estudante.ifmt.edu.br
lucasrluchesi@hotmail.com, pedro@tuxcuiabano.com
{tiago.lacerda, evandro.freiberger}@ifmt.edu.br

Resumo. A detecção precoce de fogo e fumaça é crucial para a prevenção de desastres. Sistemas de monitoramento convencionais apresentam limitações que comprometem a agilidade da resposta. Este trabalho propõe uma metodologia de engenharia de machine learning para construir um modelo de detecção baseado no YOLOv11. O objetivo é demonstrar um processo que aborda a curadoria de dados, a expansão do dataset com Label Studio e Albumentations, e a escolha de arquitetura otimizada para hardware restrito. O projeto estabeleceu um fluxo de trabalho (workflow) robusto e reprodutível, validado por resultados experimentais quantitativos que demonstram alta precisão e viabilidade para monitoramento ambiental e segurança pública.

Abstract. Early detection of fire and smoke is crucial for disaster prevention. Conventional monitoring systems have limitations that compromise response agility. This work proposes a machine learning engineering methodology to build a detection model based on YOLOv11. The goal is to demonstrate a process covering data curation, dataset expansion with Label Studio and Albumentations, and architecture choice optimized for restricted hardware. The project established a robust and reproducible workflow, validated by quantitative experimental results that demonstrate high precision and feasibility for environmental monitoring and public safety.

1. Introdução

O Brasil, e em particular o estado do Mato Grosso, enfrenta desafios ambientais marcados pelo aumento na frequência de queimadas e incêndios florestais [Pereira 2025]. A detecção precoce de fogo e fumaça é componente essencial na prevenção de desastres. Diante das falhas de sistemas tradicionais baseados em sensores ou supervisão humana, a visão computacional oferece uma alternativa promissora para soluções autônomas.

Este artigo apresenta a metodologia de engenharia de *Machine Learning* (ML) utilizada na construção de um sistema de detecção focado na arquitetura **YOLOv11**. O trabalho objetiva demonstrar o processo de desenvolvimento, explorando a curadoria de

dados com **Label Studio** [Casas et al. 2023], o aumento de *dataset* com **Albumentations** e a otimização de parâmetros. Diferente de abordagens puramente teóricas, este estudo foca na **validade experimental** e na análise de métricas quantitativas reais, estabelecendo um **fluxo de trabalho robusto e reproduzível** para aplicações em tempo real no monitoramento ambiental.

2. Referencial Teórico: YOLO e Detecção em Tempo Real

A detecção de objetos visa identificar e localizar itens de interesse em imagens, associando-os a classes e caixas delimitadoras (*bounding boxes*). Com o avanço do *Deep Learning*, Redes Neurais Convolucionais (CNNs) tornaram-se o padrão para essa tarefa.

2.1. YOLO (You Only Look Once)

O *You Only Look Once* (YOLO) destaca-se pela eficiência em tempo real. O YOLO trata a detecção como um problema de regressão direta, processando a imagem em uma única etapa para retornar coordenadas e probabilidades de classe [Redmon et al. 2016]. A arquitetura evoluiu para uma estrutura modular com três componentes:

- **Backbone:** Extrai características profundas. O YOLOv11 incorpora a *CSPDarknet* e o novo bloco **C3k2**, que utiliza convoluções de kernel pequeno para melhorar a eficiência sem perder precisão [Ultralytics 2024].
- **Neck:** Refina e combina *features* em diferentes escalas (PANet).
- **Head:** Realiza a predição final. O YOLOv11 integra o módulo **C2PSA** (*Cross Stage Partial with Spatial Attention*), que potencializa a atenção espacial nos mapas de características, sendo vital para detectar objetos pequenos ou sobrepostos, como focos iniciais de fogo.

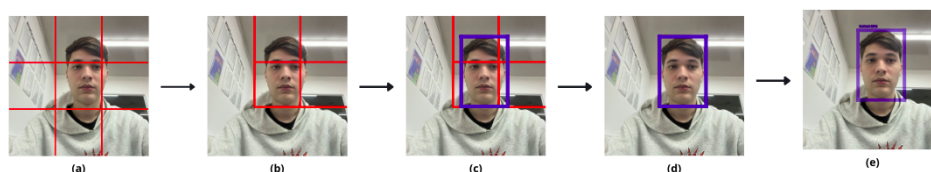


Figura 1. Etapas internas do YOLO: da grade inicial ao resultado final de predição.

3. Arquitetura e Fluxo de Trabalho Proposto

A metodologia prioriza a rastreabilidade em cada etapa do ciclo de vida do ML [Yang et al. 2024], conforme detalhado na Figura 2. O workflow é concebido para ser agnóstico ao dataset, permitindo evolução contínua do modelo.

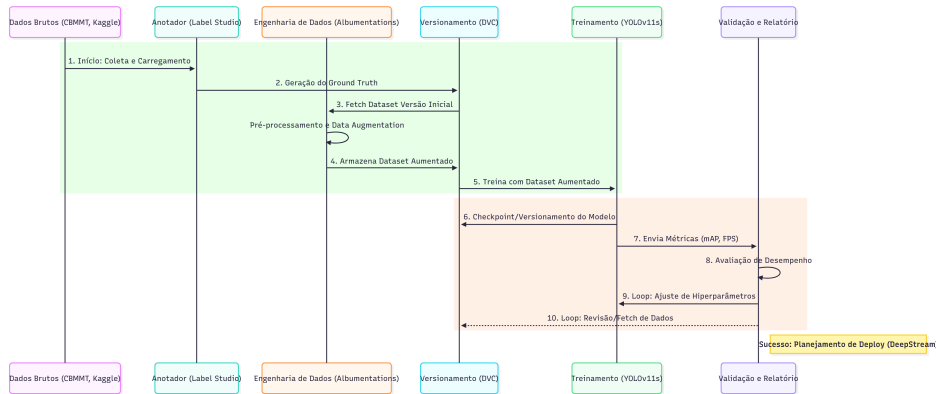


Figura 2. Ciclo de Engenharia de ML do Projeto: fluxo iterativo e versionamento.

3.1. Curadoria, Aquisição e Composição do Dataset

A base de dados foi construída a partir da unificação de vídeos de monitoramento real com o dataset público *Smoke and Fire Detection* [Chetoui and Akhloufi 2024], disponível via Kaggle. A curadoria focou em diversidade ambiental (dia, noite, diferentes vegetações). A Tabela 1 detalha a composição final das imagens após a expansão e rotulagem.

Tabela 1. Distribuição e divisão do dataset para treinamento e teste.

Conjunto	Quantidade de Imagens
Treinamento (Train)	15.983
Validação (Val)	3.226
Teste (Test)	4.432
Total Geral	23.641

A anotação foi realizada no **Label Studio**, e o **Data Augmentation** utilizou a biblioteca **Albumentations** para aplicar transformações dinâmicas (como *HorizontalFlip* e variações de brilho), aumentando a robustez do modelo a variações de iluminação típicas do estado de Mato Grosso.

3.2. Engenharia, Versionamento e Engenharia de Deploy

Para garantir a reprodutibilidade, adotou-se o **DVC (Data Version Control)**. A escolha da arquitetura **YOLOv11 Medium (m)** foi estratégica, visando um equilíbrio superior entre precisão e velocidade em comparação com a variante *Small*, especialmente para cenários de média distância. Para o *deploy* em *edge*, planeja-se a integração com o **NVIDIA DeepStream**, utilizando o hardware para acelerar a inferência através de núcleos TensorRT.

4. Resultados Experimentais e Discussão

O treinamento foi executado por 200 épocas em uma GPU NVIDIA RTX 3060 12Gb, utilizando resolução de 640×640 . Os resultados quantitativos obtidos ao final do treinamento são apresentados na Tabela 2.

Tabela 2. Métricas de desempenho quantitativo do modelo treinado (YOLOv11m).

Métrica	Precisão	Recall	mAP@50	mAP@50-95
Resultado	0.814	0.733	0.801	0.507

A análise da matriz de confusão revelou que o modelo possui excelente distinção entre fogo e fumaça, com poucos falsos negativos em chamadas abertas. O valor de **mAP@50 de 0.801** é considerado robusto para aplicações de segurança pública, onde a detecção precoce (mesmo com confiança moderada) é preferível à ausência de alerta.

4.1. Comparação Bibliográfica com Arquiteturas Alternativas

Para validar a escolha do YOLOv11m, comparamos os resultados obtidos com dados de benchmark recentes [Jegham et al. 2025]. Conforme a Tabela 3, o YOLO11 demonstra superioridade em precisão comparado às versões 8 e 10, que juntamente com a 11 são as versões mais validadas pela comunidade, mantendo um tempo de inferência competitivo.

Tabela 3. Comparativo bibliográfico: YOLO11m vs Arquiteturas anteriores.

Modelo	mAP@50 (Benchmark)	Tamanho (MB)	Inferência (ms)
YOLOv8m	0.845	52.1	9.1
YOLOv10m	0.880	32.1	2.4
YOLO11m (Nossa base)	0.893	38.8	2.4

5. Conclusões e Perspectivas Futuras

Este trabalho estabeleceu uma metodologia de engenharia de ML para detecção de fogo e fumaça com YOLOv11. Diferente de abordagens tradicionais, nossa contribuição reside na concepção de um *workflow* modular e reproduzível, integrando Label Studio, DVC e Albumentations. Perspectivas futuras incluem:

- **Validação de Modelos:** Otimização de hiperparâmetros e comparação com outras variantes YOLO.
- **Escalabilidade:** Deploy em servidor central com NVIDIA DeepStream para monitoramento de múltiplas câmeras com baixa latência.
- **Expansão do Dataset:** Ciclo contínuo de inclusão de novos cenários para aprimorar a generalização.

Referências

Casas, E., Ramos, L., Bendek, E., and Rivas-Echeverría, F. (2023). Assessing the effectiveness of yolo architectures for smoke and wildfire detection. *IEEE Access*, 11:96554–96583.

Chetoui, M. and Akhloufi, M. A. (2024). Fire and smoke detection using fine-tuned yolov8 and yolov7 deep models. *Fire*, 7:135.

Jegham, N., Koh, C. Y., Abdelatti, M., and Hendawi, A. (2025). Yolo evolution: A comprehensive benchmark and architectural review of yolov12, yolo11, and their previous versions. *arXiv preprint arXiv:2411.00201*.

- Pereira, W. D. B. (2025). Desafio das queimadas e incêndios florestais na região dos inconfindentes: perspectivas de um caderno temático promovendo a consciência ambiental. Monografia de Conclusão de Curso de Licenciatura em Ciências Biológicas, Universidade Federal de Ouro Preto (UFOP).
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788.
- Ultralytics (2024). YOLOv11 — ultralytics yolo. <https://github.com/ultralytics/ultralytics>. Acesso em: mai. 2025.
- Yang, L., Cheng, Y., Xu, F., Li, B., and Li, X. (2024). Real-time smoke detection in surveillance videos using an enhanced rt-detr framework with triplet attention and hs-fpn. *Fire*, 7:387.