Avaliação do YOLOv11 para Detecção de Árvores

Matheus Duarte da Silva
Matrícula: 21/1062277
Departamento de Ciências da Computação
Universidade de Brasília (UnB)
Brasília, Brasil
mduartesilva03@gmail.com

***Resumo*—Este trabalho avalia o desempenho de um detector de objetos moderno, o YOLOv11, na tarefa de detecção de árvores em ambientes urbanos, comparando seus resultados com o estudo de Zamboni et al., que realizou um benchmark com 21 outros modelos. O modelo YOLOv11 alcançou uma média de $AP_{50} = 0.69561$, demonstrando ser uma ferramenta eficaz e competitiva para a contagem automatizada de árvores em cenários urbanos complexos.**

***Index Terms*—Detecção de Objetos, Deep Learning, YOLO, Sensoriamento Remoto, Contagem de Árvores, Visão Computacional**

I. Introdução

O crescimento acelerado da população em áreas urbanas traz diversos desafios, especialmente quando não é acompanhado por um planejamento adequado. A expansão do espaço urbano, muitas vezes realizada de forma desordenada, tende a desconsiderar áreas naturais existentes nos arredores das cidades, o que impacta negativamente o meio ambiente e, consequentemente, a qualidade de vida da população [1]. Esses impactos podem se manifestar por meio de fenômenos como ilhas de calor, enchentes e perda de biodiversidade [2], [3]. Em regiões com presença de rios e afluentes, observa-se que o processo de urbanização contribui para a poluição hídrica e períodos de seca [4].

A arborização urbana e a preservação das áreas naturais são estratégias essenciais para a mitigação desses problemas. As árvores urbanas oferecem uma série de benefícios ambientais, como a absorção de poluentes atmosféricos e particulados, a regulação térmica e hídrica, além de proporcionarem sombra, conforto ambiental e embelezamento da paisagem. Também desempenham um papel importante na saúde física e mental da população [2], [3], [5].

Apesar disso, o inventário e o monitoramento de árvores em escala urbana ainda são tarefas desafiadoras quando realizadas manualmente, devido à grande quantidade e dispersão dos exemplares. Nesse contexto, técnicas de *Deep Learning* aplicadas a imagens aéreas de alta resolução surgem como uma alternativa promissora. Detectores de objetos com alto grau de acurácia podem automatizar parte significativa desse processo, reduzindo o esforço humano necessário para identificação e catalogação de árvores.

Zamboni et al. [6] realizaram um estudo comparativo com 21 modelos de detecção de objetos, incluindo abordagens *anchor-based* (de um e dois estágios) e *anchor-free*. Os resultados indicaram melhor desempenho médio dos modelos *anchor-free*, com destaque para o FSAF, que obteve $AP_{50} = 0.701$. Métodos amplamente utilizados como Faster R-CNN e RetinaNet apresentaram desempenho inferior.

Em 2024, a empresa Ultralytics lançou o **YOLOv11**, uma nova versão da conhecida família de detectores de objetos *You Only Look Once*, com foco em desempenho em tempo real e precisão *state-of-the-art* em múltiplas tarefas de visão computacional [7], [8].

Este trabalho investiga a aplicação do YOLOv11 na detecção de copas de árvores individuais no conjunto de dados urbanos de Campo Grande (Brasil), utilizado também por Zamboni et al. O objetivo é avaliar se um modelo moderno de estágio único (*one-stage*) pode superar os resultados de modelos *anchor-free* anteriores.

As principais contribuições deste trabalho são:

- adaptação do pipeline experimental de [6] para o modelo YOLOv11 [8];
- análise quantitativa baseada em validação cruzada 5-fold, com comparação dos resultados obtidos com os da literatura.

O restante do artigo está organizado da seguinte forma: a Seção II discute trabalhos relacionados; a Seção III descreve os dados e a metodologia; a Seção IV apresenta e analisa os resultados; e a Seção V conclui com apontamentos para trabalhos futuros.

II. Trabalhos Relacionados

A detecção automatizada de árvores individuais tornou-se uma área de pesquisa relevante em aplicações ambientais, agrícolas e de planejamento urbano. O avanço de técnicas de *Deep Learning*, aliado à crescente disponibilidade de imagens aéreas de alta resolução, tem impulsionado o desenvolvimento de soluções robustas para este desafio. Entre as arquiteturas mais investigadas, destacam-se os modelos de detecção de objetos como Faster R-CNN, RetinaNet e as diversas variantes da família YOLO (You Only Look Once).

Neste contexto, o trabalho de Zamboni et al. [6] — base para este projeto — realizou uma análise comparativa extensiva de 21 métodos, estabelecendo um importante benchmark para a detecção de árvores em ambientes urbanos. Para contextualizar a nossa contribuição, analisamos a seguir outros estudos que abordam desafios semelhantes.

Beloiu et al. [9] exploraram o uso da arquitetura Faster R-CNN para detectar e classificar quatro espécies arbóreas em florestas temperadas na Suíça, utilizando imagens aéreas RGB. O estudo demonstrou que a abordagem multi-espécies foi eficaz mesmo em cenários com grande sobreposição de copas e iluminação variada, alcançando um F1-score de 0.92 para a espécie minoritária *P. sylvestris* e mantendo um desempenho estável para as demais. A pesquisa reforça a viabilidade de modelos de dois estágios para mapear espécies em florestas com estruturas complexas.

Em um cenário agrícola, Hnida et al. [10] propuseram um modelo avançado para a detecção multiescala de copas de oliveiras em ortofotos de alta resolução capturadas por VANTs (Veículos Aéreos Não Tripulados). A arquitetura, que incorporou componentes como CSPNet, FPN e PAN, alcançou um notável mAP_{50} de 94,0% e F1-score de 91,3%. Embora focado na agricultura de precisão, o estudo aborda desafios de detecção — como variações de escala e sobreposição de objetos — que são diretamente análogos aos encontrados em ambientes urbanos.

Os trabalhos mencionados demonstram o sucesso de arquiteturas complexas em cenários específicos. No entanto, há uma necessidade contínua de avaliar modelos mais recentes e eficientes, como os da família YOLO, que são projetados para otimizar o balanço entre velocidade e acurácia. Este estudo se insere nesta lacuna, propondo a avaliação do YOLOv11 [7], [8], [11], um detector de estágio único de alta eficiência. Ao aplicá-lo ao benchmark de Zamboni et al., investigamos se esta nova arquitetura pode oferecer um desempenho competitivo ou superior aos métodos do estado da arte já analisados para a arborização urbana.

III. Metodologia

Esta seção detalha os procedimentos experimentais adotados, incluindo a descrição do conjunto de dados, a arquitetura do modelo, a configuração de treinamento e as métricas utilizadas para avaliar a performance.

A. Conjunto de Dados

O estudo utilizou o conjunto de dados público disponibilizado por Zamboni et al. [6]. Originalmente, os dados consistem em duas ortoimagens RGB da área urbana de Campo Grande, Brasil, com um *Ground Sample Distance* (GSD) de 10 cm. Para os experimentos, estas imagens foram particionadas em 220 patches de 512 × 512 pixels, resultando em uma área de 2621,44 m² por patch.

Ao todo, **3.382 copas de árvores** foram manualmente anotadas com retângulos delimitadores (*bounding boxes*) e utilizadas como ground-truth. O dataset foi disponibilizado com uma estrutura de validação cruzada de 5-folds, que foi integralmente adotada neste trabalho para garantir uma avaliação robusta e comparável. Antes do treinamento, as anotações foram convertidas para o formato YOLO, que consiste em um ficheiro de texto por imagem contendo as coordenadas normalizadas $(x_{centro}, y_{centro}, largura, altura)$ de cada objeto.

B. Modelo Utilizado

O modelo selecionado para este trabalho foi o **YOLOv11-nano**, uma arquitetura da família *You Only Look Once*. Os modelos YOLO são detectores de estágio único (*one-stage*) reconhecidos por sua alta eficiência, equilibrando velocidade de inferência e acurácia [11]. A versão *nano* é a mais leve da arquitetura, tornando-a ideal para aplicações que demandam processamento rápido ou que são executadas em hardware com recursos limitados [11].

A arquitetura do YOLOv11 incorpora otimizações notáveis, como o módulo C2PSA (*Convolutional block with Parallel Spatial Attention*), que aprimora a capacidade de extração de características espaciais (*features*) dos objetos [7], [11]. Estudos indicam que, além de sua versatilidade em diferentes tamanhos (do nano ao extra-large), o YOLOv11 apresenta ganhos de performance e eficiência computacional sobre suas versões anteriores [11].

Neste trabalho, um modelo pré-treinado na base de dados COCO (`yolo11n.pt`) foi ajustado (*fine-tuned*) para a tarefa específica de detecção de árvores, utilizando a biblioteca `Ultralytics` em um ambiente Google Colab com utilização de GPU.

C. Configuração Base

Para garantir a robustez dos resultados, foram realizadas cinco execuções de treinamento distintas, uma para cada fold da validação cruzada. Os principais hiperparâmetros foram mantidos consistentes em todas as execuções, alinhados à metodologia do estudo de referência [6]:

- **Épocas**: 24
- **Tamanho da imagem (Input Size)**: 512×512 pixels
- **Otimizador e Taxa de Aprendizagem**: Padrão da biblioteca `Ultralytics` para garantir a reprodutibilidade.

D. Métricas de Avaliação

A avaliação da performance do modelo seguiu o protocolo padrão para tarefas de detecção de objetos, baseado no cálculo da *Average Precision* (AP). Para compreender a AP, é necessário primeiro definir os conceitos de Precisão e Recall.

Uma detecção é classificada com base no seu *Intersection over Union* (IoU) — a razão entre a área de sobreposição e a área de união da *bounding box* predita e da anotação real. Dado um limiar de IoU (ex: 0.5), temos:

- **True Positive (TP)**: Uma detecção correta, onde o IoU é superior ao limiar.
- **False Positive (FP)**: Uma detecção incorreta (IoU abaixo do limiar ou detecção em uma área sem objeto).
- **False Negative (FN)**: Um objeto real que o modelo não conseguiu detectar.

A partir destes componentes, a **Precisão (Precision)** e o **Recall** são definidos. A Precisão mede a acurácia das predições feitas pelo modelo, enquanto o Recall mede a capacidade do modelo de encontrar todos os objetos existentes. Suas fórmulas são:

$$\text{Precisão} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

A **Average Precision (AP)** para uma classe de objeto é calculada como a área sob a curva Precisão-Recall, que plota a precisão em função do recall para todos os limiares de confiança. Ela fornece uma métrica única e robusta da performance do modelo.

As métricas específicas utilizadas neste trabalho foram:

- **AP**$_{50}$: A AP calculada com um limiar de IoU fixo em 0.5. É a métrica mais comum para comparação em benchmarks de detecção de objetos.
- **AP**$_{50\text{-}95}$: Uma métrica mais rigorosa, que representa a média da AP calculada sobre 10 limiares de IoU, variando de 0.5 a 0.95, com um passo de 0.05.

Os resultados finais foram consolidados calculando-se a média e o desvio padrão de ambas as métricas ao longo dos cinco folds da validação cruzada.

IV. Resultados e Discussão

A análise a seguir detalha os resultados quantitativos, comparando-os com o benchmark de referência, e discute o comportamento do modelo durante o processo de treinamento.

A. Análise Quantitativa e Comparativa

Para cada um dos cinco folds, o modelo foi treinado por 24 épocas e avaliado em seu respectivo conjunto de teste. A Tabela I sumariza os valores de performance obtidos para as métricas de AP_{50} e $AP_{50\text{-}95}$.

Tabela I
Resultados de Performance do YOLOv11n por Fold

Fold	AP$_{50}$	AP$_{50\text{-}95}$
0	0.690	0.370
1	0.696	0.360
2	0.686	0.360
3	0.715	0.380
4	0.686	0.353
Média	**0.695**	**0.365**
Desvio Padrão	**0.012**	**0.010**

O modelo alcançou um AP_{50} médio de **0.695** $\pm$ **0.012**, um resultado que demonstra alta consistência entre os diferentes subconjuntos de dados. Este valor posiciona o YOLOv11n como um dos modelos de melhor desempenho, ficando apenas **0.8%** abaixo do líder do benchmark, o modelo FSAF ($AP_{50} = 0.701$). Notavelmente, a performance do YOLOv11n supera a média dos modelos *anchor-free* ($AP_{50} = 0.686$) e também o seu predecessor, o YOLOv3 ($AP_{50} = 0.591$), em mais de 10 pontos percentuais, apontando um avanço geracional significativo [6].

O resultado médio de $AP_{50\text{-}95}$ de **0.365** indica que o modelo mantém uma boa precisão de localização mesmo em limiares de IoU mais rigorosos, sugerindo que as caixas delimitadoras preditas são bem ajustadas aos objetos reais.

B. Análise do Comportamento do Treino e Discussão

A escolha de treinar por 24 épocas, alinhada à metodologia do artigo base, provou ser suficiente para alcançar resultados competitivos. A análise das curvas de treino e validação, exemplificadas na Fig. 1 para um dos folds, mostra um padrão consistente em todas as cinco execuções. As curvas de perda (*loss*) decrescem rapidamente e as métricas de validação, como o AP_{50}, sobem de forma acentuada nas primeiras 15 épocas, indicando um aprendizado eficiente.

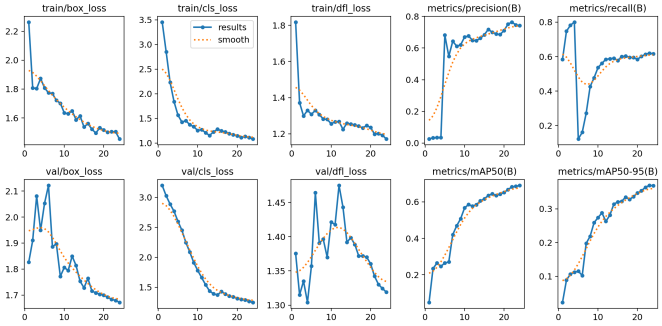

Figura 1. Curvas de treino e validação para o Fold 0

Embora as curvas comecem a estabilizar, elas não atingem um platô completo, sugerindo que um treinamento mais longo poderia refinar o modelo e, potencialmente, levar a ganhos incrementais de performance.

Uma análise mais profunda da relação entre o score F1 e o limiar de confiança do modelo é apresentada na Fig. 2. Consistentemente entre os folds, o pico de performance F1 (entre 0.66 e 0.69) é alcançado com um limiar de confiança relativamente baixo, em torno de 0.27. Esta informação é valiosa para otimizar a aplicação prática do modelo, pois permite definir um limiar de decisão que equilibra de forma ideal a precisão e o recall das detecções.

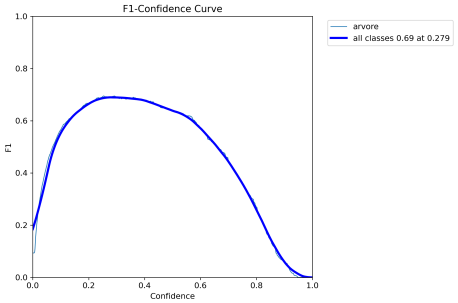

Figura 2. Curva F1-Confiança para o Fold 3

Em suma, a discussão quantitativa e a análise do comportamento do treino revelam que o YOLOv11n não é apenas competitivo em termos de acurácia, mas também um modelo bem-comportado e previsível. Apesar de sua leveza, ele demonstrou uma capacidade notável de aprender características complexas das copas das árvores, rivalizando com arquiteturas mais pesadas e se estabelecendo como uma ferramenta poderosa e eficiente para aplicações de sensoriamento remoto urbano.

C. Discussão e Análise Qualitativa

Um dos achados mais relevantes deste estudo é a demonstração de que a performance de detecção do YOLOv11n, conforme detalhado na seção anterior, é acompanhada por uma notável eficiência computacional. Apesar de ser uma arquitetura leve, com 100 camadas, aproximadamente 2.6 milhões de parâmetros e 6.5 GFLOPs de custo computacional [7], [8], [11], o modelo demonstrou um desempenho quantitativo que rivaliza com arquiteturas mais complexas.

Esta eficiência é confirmada pelo tempo de inferência, que foi medido para cada fold no conjunto de teste em uma GPU Tesla T4. A Tabela II apresenta estes valores, que resultaram em uma média de apenas **9.74 ms** por imagem.

Tabela II
TEMPO DE INFERÊNCIA DO YOLOV11N POR IMAGEM EM CADA FOLD

Fold	Tempo de Inferência (ms)
0	12.5
1	15.3
2	8.0
3	6.8
4	6.1
Média	**9.74**
Desvio Padrão	**3.98**

A combinação de alta acurácia (AP_{50} de 0.695) com baixa latência de inferência torna o YOLOv11n um forte candidato para aplicações em larga escala e em tempo real, onde o processamento rápido de grandes volumes de imagens é um requisito fundamental. É importante notar a variabilidade no tempo de inferência entre os folds, indicada pelo desvio padrão de 3.98 ms. Essa flutuação pode ser atribuída tanto a variações no ambiente de execução quanto à complexidade distinta das imagens em cada conjunto de teste.

Do ponto de vista qualitativo, a análise das predições revela que o modelo é particularmente eficaz na detecção de árvores isoladas ou em grupos de baixa densidade, independentemente do tamanho da copa. No entanto, o principal desafio, em linha com as observações do estudo de referência [6], reside em áreas de alta densidade de vegetação, onde as copas se sobrepõem. Nestes cenários complexos, o modelo por vezes agrupa múltiplas árvores em uma única detecção ou falha em identificar árvores menores sob a sombra de outras maiores.

D. Limitações e Trabalhos Futuros

Embora os resultados deste estudo sejam promissores, é importante reconhecer suas limitações, que, por sua vez, abrem caminhos para pesquisas futuras.

As principais limitações identificadas são:

- **Escopo da Arquitetura:** A avaliação se restringiu à variante "nano"do YOLOv11. Embora eficiente, ela pode não representar o potencial máximo de acurácia da família de modelos.
- **Rigor Estatístico:** Não foram realizadas análises estatísticas formais, como o teste ANOVA com correção de Holm-Bonferroni, para determinar se as diferenças de performance entre o YOLOv11n e os modelos de topo do benchmark são estatisticamente significativas.
- **Medição de Eficiência:** O tempo de inferência foi medido em um ambiente de nuvem (Google Colab), que pode apresentar flutuações. Um benchmark formal em hardware padronizado seria necessário para uma avaliação de eficiência mais rigorosa.

Com base nestes pontos, os seguintes trabalhos futuros são propostos:

- Avaliar o desempenho de variantes mais robustas do YOLOv11 (e.g., YOLOv11-s ou YOLOv11-m) para quantificar o *trade-off* entre acurácia e custo computacional.
- Realizar um benchmark de eficiência mais detalhado, incluindo métricas como FPS (Frames Per Second) e latência em diferentes hardwares.
- Estender a análise para outros ambientes urbanos e tipos de vegetação, utilizando novos conjuntos de dados para avaliar a capacidade de generalização e a robustez do modelo.

V. CONCLUSÃO

Este trabalho se propôs a avaliar a eficácia do detector de objetos YOLOv11n para a tarefa de detecção de árvores individuais em imagens aéreas urbanas, contextualizando sua performance frente ao extensivo benchmark realizado por Zamboni et al. [6]. Através de uma metodologia rigorosa de validação cruzada de 5-folds e mantendo consistência com os parâmetros do estudo de referência, foi possível posicionar esta nova arquitetura no estado da arte para a aplicação.

Os resultados demonstraram que, mesmo em sua versão mais leve ("nano"), o YOLOv11n alcançou um desempenho notável, com um AP_{50} médio de **0.695**. Este valor não apenas supera a média dos modelos avaliados no estudo original, como também rivaliza diretamente com as melhores arquiteturas, provando ser uma solução altamente competitiva em termos de acurácia. Além disso, o modelo se destacou por sua alta eficiência computacional, com um tempo de inferência médio de apenas 9.74 ms, um fator crucial para aplicações em larga escala.

A principal contribuição deste estudo é a demonstração de que arquiteturas modernas e eficientes de estágio único, como o YOLOv11, podem atingir um patamar de performance comparável a modelos mais complexos, sem sacrificar a velocidade de processamento. Isto sinaliza um avanço importante para o monitoramento ambiental urbano, viabilizando o desenvolvimento de ferramentas automatizadas que são, ao mesmo tempo, precisas e rápidas.

Como trabalhos futuros, recomenda-se a avaliação de variantes mais robustas da família YOLOv11, a realização de benchmarks de eficiência em hardware padronizado e a expansão da análise para conjuntos de dados de diferentes cidades, a fim de aprimorar a capacidade de generalização dos modelos para a gestão de ecossistemas urbanos em escala global.

Referências

[1] B. Güneralp and K. C. Seto, "Environmental impacts of urban growth from an integrated dynamic perspective: A case study of shenzhen, south china," *Global Environmental Change*, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0959378008000587

[2] F. Angeoletto, T. E. P. N. Duarte, J. W. M. C. Santos, F. F. Silva, J. F. C. Bohrer, and L. Massad, "Reflexões sobre arborização urbana: desafios a serem superados para o incremento da arborização urbana no Brasil," *Revista em Agronegócio e Meio Ambiente*, 2018. [Online]. Available: https://www.researchgate.net/publication/324068470_REFLEXOES_SOBRE_ARBORIZACAO_URBANA_DESAFIOS_A_SEREM_SUPERADOS_PARA_O_INCREMENTO_DA_ARBORIZACAO_URBANA_NO_BRASIL

[3] Universidade Federal de Santa Maria. (2024) A importância da arborização urbana para cidades sustentáveis. Acesso em: 04 jul. 2025. [Online]. Available: https://www.ufsm.br/unidades-universitarias/ccne/2024/06/20/a-importancia-da-arborizacao-urbana-para-cidades-sustentaveis

[4] R. C. da Silva Menezes e George Rembrandt Gutlich. (2015) Apontamentos do crescimento urbano e o desafio da preservação ambiental. Acesso em: 04 jul. 2025. [Online]. Available: https://www.revistaespacios.com/a16v37n08/16370810.html

[5] C. T. C. e Samara Simon Christmann e Tarcísio Dorn de Oliveira. (2014) Arborização urbana: Importância e benefícios no planejamento ambiental das cidades. Acesso em: 04 jul. 2025. [Online]. Available: https://tinyurl.com/2485uhar

[6] P. Zamboni, J. M. Junior, J. de Andrade Silva, G. T. Miyoshi, E. T. Matsubara, K. Nogueira, and W. N. Gonçalves, "Benchmarking Anchor-Based and Anchor-Free State-of-the-Art Deep Learning Methods for Individual Tree Detection in RGB High-Resolution Images," *Remote Sensing*, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/13/2482

[7] Ultralytics. (2024) Yolov11 documentation. Acesso em: 04 jul. 2025. [Online]. Available: https://docs.ultralytics.com/models/yolo11/

[8] G. Jocher and J. Qiu, "Ultralytics yolo11," 2024. [Online]. Available: https://github.com/ultralytics/ultralytics

[9] M. Beloiu, L. Heinzmann, N. Rehush, A. Gessler, and V. C. Griess, "Individual tree-crown detection and species identification in heterogeneous forests using aerial rgb imagery and deep learning," *Remote Sensing*, 2023. [Online]. Available: https://www.mdpi.com/2072-4292/15/5/1463

[10] Y. Hnida, M. A. Mahraz, A. Yahyaouy, A. Achebour, J. Riffi, and H. Tairi, "Enhanced multi-scale detection of olive tree crowns in uav orthophotos using a deep learning architecture," *Smart Agricultural Technology*, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772375525003582

[11] R. Khanam and M. Hussain, "Yolov11: An overview of the key architectural enhancements," 2024. [Online]. Available: https://arxiv.org/abs/2410.17725

Apêndice A
Repositório do Projeto

O código-fonte, os experimentos e os arquivos utilizados neste trabalho estão disponíveis no seguinte repositório público:

`github.com/smmstakes/iia-trabalho-2`

O repositório contém:

- Notebook para treinamento e validação do YOLOv11;
- Scripts de preparação do conjunto de dados;
- Resultado dos treinamentos com seus gráficos e demais informações;
- Arquivo `README.md` com instruções para reprodução.